

การศึกษาการสร้างโปรแกรมจำลองการทำงานของ
ของหุ่นยนต์เคลื่อนที่อิสระ 6 แกน

A Study of Program Six Axis Robot Kinematics Simulation

ชนะ รัชศิริ^{1,2*} จาตุรนต์ รอดอ่วม¹ วโรดม ตูจินดา^{1,3} และชัยยากร จันทร์สุวรรณ^{1,3}
¹ศูนย์เชี่ยวชาญเฉพาะทางแม่พิมพ์ยาง สถาบันค้นคว้าและพัฒนาเทคโนโลยีการผลิตทางอุตสาหกรรม
²ภาควิชาวิศวกรรมอุตสาหกรรม คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเกษตรศาสตร์ และ
³ภาควิชาวิศวกรรมเครื่องกล คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเกษตรศาสตร์ เขตจตุจักร กรุงเทพฯ 10900
โทร 0-2942-7188 โทรสาร 0-2942-7189 *อีเมลล์ chana_raksiri@yahoo.com

Chana Raksiri^{1,2*} Jaturon Rodauam¹ Varodom Tujinda^{1,3} and Chiyachon Jansuwan^{1,3}

¹Center of Excellence in Rubber Mould, Research and Development Institute of Production Technology

²Department of Industrial Engineering, Faculty of Engineering, Kasetsart University

³Department of Mechanical Engineering, Faculty of Engineering, Kasetsart University, Jatujak, Bangkok 10900, Thailand,

Tel: 0-2942-7188 Fax 0-2942-7189 *E-mail: chana_raksiri@yahoo.com

บทคัดย่อ

การศึกษานี้ เป็นการศึกษเกี่ยวกับการจำลองการทำงานของ หุ่นยนต์เคลื่อนที่อิสระ 6 แกน ก่อนที่จะมีการนำไปใช้งานกับหุ่นยนต์จริง โดยใช้วิธีแบบจำลองกราฟิก (Graphic Simulator) มาใช้ในการตรวจสอบการเคลื่อนที่ของหุ่นยนต์ ทำให้ผู้ใช้สามารถตรวจสอบข้อผิดพลาดต่าง ๆ ที่เกิดขึ้นได้จากโปรแกรมนี้ และทำการจำลองเส้นทางการเดินใหม่ได้ทันที โปรแกรมจำลองการทำงานของหุ่นยนต์เคลื่อนที่อิสระ 6 แกนนี้ สามารถจำลองการทำงานของหุ่นยนต์ Fanuc Robot S-5 โดยใช้ Microsoft Visual Basic 6.0 และ OpenGL ในการเขียนโปรแกรมร่วมกับข้อมูลโปรแกรมบางส่วนที่ได้มา แล้วนำมาพัฒนาและประยุกต์ใช้กับการค้นคว้าโปรแกรมนี้ จากการค้นคว้าและพัฒนาโปรแกรมสามารถแสดงให้เห็นการทำงานของหุ่นยนต์เคลื่อนที่อิสระ 6 แกน Fanuc Robot-S5 โดยแสดงกราฟิกหุ่นยนต์เคลื่อนที่ตามข้อมูลที่ป้อนให้ แสดงค่าพิกัด X,Y,Z ของปลายแขนหุ่นยนต์ (Absolute Position) มีการเก็บข้อมูลการเคลื่อนที่เป็นชุดคำสั่งพร้อมแสดงตารางข้อมูล สามารถนำไปพัฒนาเพิ่มแบบจำลอง การต่อโยง ของกลไกหุ่นยนต์รุ่นอื่น และยังสามารถนำมาเป็น สื่อในการเรียนการสอนในวิชาการหุ่นยนต์เพื่อให้ผู้ศึกษาเข้าใจเกี่ยวกับพื้นฐานวิชาการหุ่นยนต์ได้ดียิ่งขึ้น

คำสำคัญ : หุ่นยนต์เคลื่อนที่อิสระ 6 แกน, กราฟิกไลบรารี

Abstract

This research of Program Six Axis Robot Kinematics simulation . It's model before using with real robot. Graphic Simulator is used for moved checking, So user can check some mistake and modify the new route instantly. A Study of Program Six Axis Robot Kinematics Simulation can model Fanuc Robot s-5'working Microsoft Visual Basic 6.0 and OpenGL program to writing program in this project. More over it join with some program and then we develop and apply it with research of this program. From this research and development, the program can be shown the working of Fanuc Robot s-5. It be shown graphics of moving robot by the inputted data and show limit x, y, z on absolute Position. Then it saves data set of movement and shown data form. It can be developed to as the link's model of new machine in other robot. More over it can be the study media of concepts of robot, So student can well understand the foundation of the basic concepts of robotics.

Keywords : Six Joints Robot, OpenGL

1. บทนำ

ในปัจจุบันงานผลิตและงานประกอบชิ้นส่วน ในภาคอุตสาหกรรมต่างๆ โดยเฉพาะอย่างยิ่งในงานอุตสาหกรรมผลิตชิ้นส่วนรถยนต์ เครื่องอุปโภคบริโภค ฯลฯ มีการใช้ หุ่นยนต์เคลื่อนที่อิสระ(Robot)เข้ามามีบทบาทในการผลิตอย่างมาก เพื่อเพิ่มประสิทธิภาพในการทำงาน เนื่องจากการทำงานที่ต้องมีการเคลื่อนที่ซับซ้อนและรวดเร็ว ทำให้มีปัญหาจากการทำงานหลายอย่าง โดยเฉพาะปัญหาการชนกันระหว่างหุ่นยนต์กับชิ้นงานและอุปกรณ์จับยึดต่างๆ และเมื่อพัฒนาชุดโปรแกรมที่ใช้จำลองการทำงานของ หุ่นยนต์เคลื่อนที่อิสระ 6 แกน บนเครื่องคอมพิวเตอร์ ทำให้สามารถตรวจสอบการชนระหว่างหุ่นยนต์เคลื่อนที่อิสระกับชิ้นงานและอุปกรณ์จับยึด ก่อนทำการผลิต และ ทำให้สามารถทราบปัญหาและแก้ไขโปรแกรมก่อน การปฏิบัติงานจริง โปรแกรมนี้จะช่วยลดความเสี่ยงการเกิดความเสียหายอันจะเกิดจากการปฏิบัติงานจริงและเป็นการศึกษากลไกการเคลื่อนที่ของ หุ่นยนต์เคลื่อนที่อิสระ 6 แกน

โปรแกรมการจำลองหุ่นยนต์ได้ถูกพัฒนาออกมาแล้วเป็นจำนวนมากในต่างประเทศ แต่แบบจำลองส่วนใหญ่ต้องใช้ผู้ที่มีความรู้ความชำนาญมาก อีกทั้งแบบจำลองที่ได้สร้างขึ้นสามารถแสดงการเคลื่อนที่ของหุ่นยนต์ได้เพียงเฉพาะรุ่น หรือทำการแก้ไขดัดแปลงรูปแบบของหุ่นยนต์ได้ยาก ดังนั้นการศึกษาวิจัย จึงมีวัตถุประสงค์ ศึกษาและสร้างโปรแกรมแบบจำลองการเคลื่อนที่ของหุ่นยนต์เคลื่อนที่อิสระ 6 แกน ของ Fanuc Robot S Model 5 ที่สามารถนำไปพัฒนาเพิ่มแบบจำลองการต่อโยง ของกลไกหุ่นยนต์รุ่นอื่น และยังสามารถนำมาเป็นสื่อในการเรียนการสอนในวิชาการหุ่นยนต์เพื่อให้ผู้ศึกษาเข้าใจเกี่ยวกับพื้นฐานวิชาการหุ่นยนต์ได้ดียิ่งขึ้น

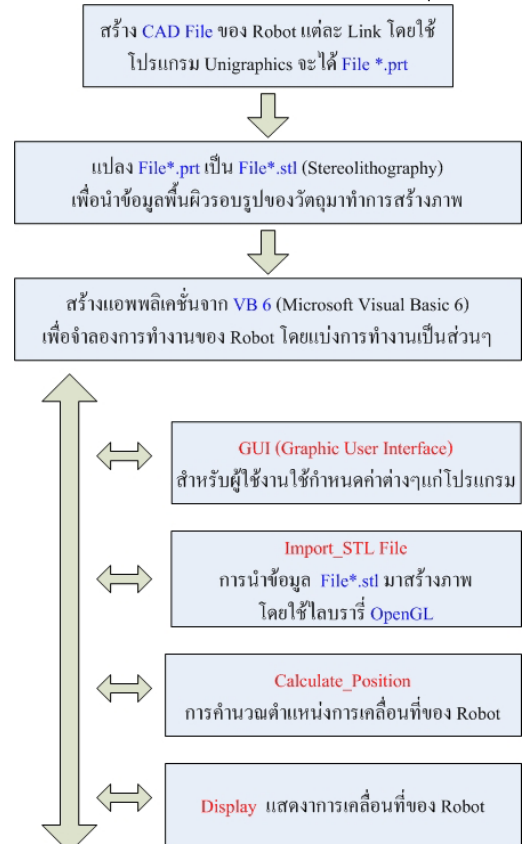
2. วิธีดำเนินงานวิจัย

โปรแกรมการจำลองการเคลื่อนที่ของหุ่นยนต์นี้ ถูกพัฒนาด้วย Microsoft Visual Basic 6 เริ่มจากการนำไฟล์ *.STL ซึ่งเป็นไฟล์มาตรฐานสำหรับระบบ CAD/CAM มีการเก็บข้อมูลพื้นที่ผิวของชิ้นงานเข้ามาสู่ไลบรารี OpenGL, GLU และ GLUT ช่วยในการเขียน และแสดงภาพกราฟิกตัวหุ่นยนต์เคลื่อนที่อิสระ 6 แกน



รูปที่ 1 Fanuc Robot S-5 ภาพจริง และ ภาพ จำลอง 3D

จากรูปที่ 1 เปรียบเทียบให้เห็นถึงการสร้างแบบจำลองของหุ่นยนต์เคลื่อนที่อิสระ 6 แกน Fanuc Robot S-5 กับหุ่นยนต์จริง



รูปที่ 2 ขั้นตอนการสร้างโปรแกรม

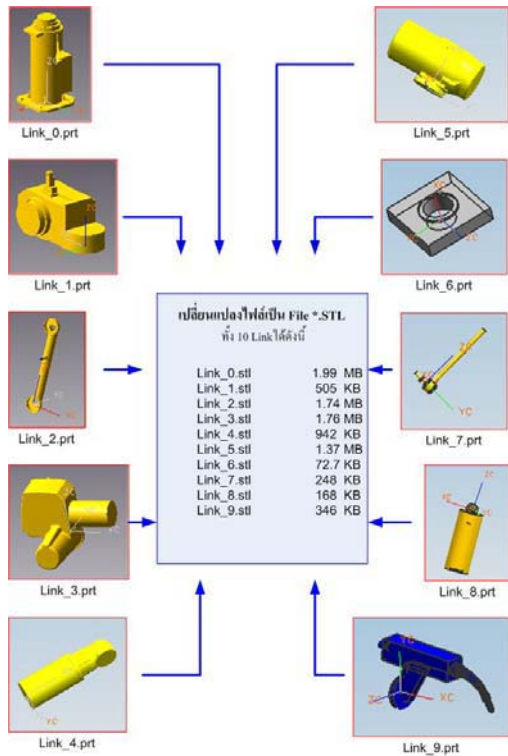
ในการพัฒนาโปรแกรมการจำลองหุ่นยนต์นี้จะมีขั้นตอนการออกแบบการพัฒนาโปรแกรมดังแสดงในรูปที่2

2.1 สร้างแบบจำลอง (CAD FILE) และเปลี่ยนแปลงชนิดข้อมูลเป็นชนิด FILE *.STL

จำนวนชิ้นส่วนของหุ่นยนต์สามารถแบ่งได้เป็น 10 ชิ้นส่วน โดยเริ่มต้นต้องสร้างไฟล์ CAD ขึ้นมาทั้ง 10 ไฟล์ โดยสร้างจาก โปรแกรม Unigraphics NX 1.0 แล้วแปลงไฟล์ เป็น FILE *.STL เพื่อนำข้อมูลเหล่านั้นมาสร้างเป็นพื้นที่ผิวของหุ่นยนต์



รูปที่ 3 เปรียบเทียบข้อมูลแต่ละชิ้นส่วนของหุ่นยนต์ จริงกับ Link ที่ได้จำลองขึ้น



รูปที่ 4 แสดงLINK ทั้งหมด จาก File *.PRT เป็น File *.STL

2.2 กำหนดจุดต่อโยงของ Robot โดยการทำหน้าที่ FILE*.DAT

นำข้อมูลพื้นผิวและทิศทางการหมุนที่ได้จาก File *.STL มา กำหนดค่าจุดต่อโยงของแต่ละ Link โดยกำหนดค่าต่างๆที่ File *.DAT File *.DAT จะทำหน้าที่กำหนด ตำแหน่ง จุดต่อโยง และการเคลื่อนที่ ของElement ต่างๆ ดังตารางข้างล่างแสดงข้อมูลต่าง ของ Element_0 ถึง Element_2 เท่านั้น

ตารางที่ 1 แสดงFunction ต่าง ๆ ของ Element_0 ถึง Element_2

Function	Element_0	Element_1	Element_2	Explain
Name	Socle	Base	Bras1	ชื่อของ Element
File	Link_0.stl	Link_1.stl	Link_2.stl	ไฟล์ของ Element
Type_axe	0	1	1	0=Fixed , 1=Move
Color	6	6	6	Color 6=สีเหลือง
Mini_axe	0	-150	-45	องศาหมุนตามเข็มนาฬิกา
Maxi_axe	0	150	65	องศาหมุนทวนเข็มนาฬิกา
Origin_x	0	0	187.563	ตำแหน่งแกน X
Origin_y	0	0	115	ตำแหน่งแกน Y
Origin_z	0	618	87.593	ตำแหน่งแกน Z
Vecter_x	0	0	0	บอกทิศทางของพื้นผิว
Vecter_y	0	0	1	
Vecter_z	0	1	0	

3. สร้างแอปพลิเคชันจาก VB 6 (Microsoft Visual Basic 6)

การพัฒนาโปรแกรมจำลองการทำงานของ Robot โดยอาศัย OpenGL ผู้พัฒนาต้องมีความรู้พื้นฐานการเขียนโปรแกรม VB 6 มาบ้าง และต้องทำการศึกษาเพิ่มเติมกับการใช้ OpenGL อีกด้วย การพัฒนาโปรแกรมนี้ Source Code จะอธิบายไว้ที่ภาคผนวก ดังนั้นจะแสดงส่วนต่างของโปรแกรม ดังนี้

3.1 GUI (Graphic User Interface)

สำหรับผู้ใช้งานใช้กำหนดค่าต่างๆ ให้แก่โปรแกรมเพื่อที่โปรแกรมจะได้ นำค่ามาคำนวณการเคลื่อนไหวของ Robot โดยการกำหนดค่าเป็นการพิมพ์ข้อมูลทั้งที่เป็นตัวเลขหรือตัวอักษร หรือการคลิกแล้วลากเมาส์ เพื่อติดต่อกับส่วนของการประมวลผล สามารถแบ่งหน้าที่การรับคำสั่งได้ดังนี้

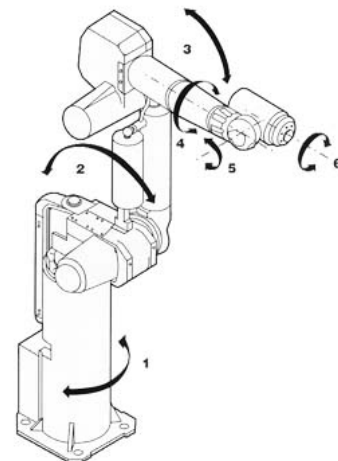
- 3.1.1 ตำแหน่งองศาการเคลื่อนที่แต่ละ Joint
- 3.1.2 ลักษณะการแสดงผลภาพ
- 3.1.3 ความเร็วการเคลื่อนที่
- 3.1.4 การพิมพ์ข้อมูลคำสั่งการเคลื่อนที่
- 3.1.5 ใช้เมาส์ควบคุมการแสดงผลทางกราฟฟิค

3.2 Import STL File

การนำข้อมูล File*.stl มาสร้างภาพ โดยใช้ข้อมูลที่กำหนดจาก *.DAT มาสร้างจุดต่อโยงของแต่ละ Link โดยใช้ไลบรารี OpenGL ทำหน้าที่สร้างกราฟฟิค เมื่อต้องการเปลี่ยน Model Robot จะต้องเปลี่ยน File*.stl และแก้ไขข้อมูลใน File*.DAT เพียงเท่านั้น โปรแกรมจะทำหน้าที่สร้างองค์ประกอบของการติดต่อกับผู้ใช้ให้เอง

3.3 การเคลื่อนที่ของหุ่นยนต์

หุ่นยนต์จะมีข้อต่อทั้งหมด 6 Joint และการเคลื่อนที่ของหุ่นยนต์รุ่นนี้จะเป็นแบบ Revolute



รูปที่ 5 แสดง ขนาดรูปทรงและทิศทางการเคลื่อนที่ของ Robot S-5

ตารางที่ 2 แสดงคุณสมบัติของ Fanuc Robot S-5

Simultaneously controlled axes		6
Motion Range and Speed	Axis 1	±150° @ 115°/sec
	Axis 2	-45° to +65° @ 90°/sec
	Axis 3	-120° to +125° @ 115°/sec
	Axis 4	±190° @ 240°/sec
	Axis 5	±140° @ 240°/sec
	Axis 6	±270° @ 400°/sec
Payload at wrist		5kg (11 lbs)
Repeatability		±0.1mm (±0.004")
Mounting method		Floor
Drive method		AC Digital Servo
Position detection system		Absolute pulse coder
Mechanical unit weight		100 kg (220 lbs)
Input power		2.5 KVA

3.3.1 คำสั่งที่ใช้ในการสร้างการเคลื่อนที่

ในการสร้างการเคลื่อนที่ของหุ่นยนต์ ส่วนของโปรแกรมจะรับข้อมูลค่ามุมมองของแต่ละ แกนจากผู้ใช้ นำมาคำนวณเพื่อ กำหนดการเคลื่อนที่ที่พิกัด X, Y, Z และสร้างภาพกราฟฟิคการเคลื่อนที่ของหุ่นยนต์

คำสั่งภายในโปรแกรม จะใช้ข้อมูลโปรแกรมที่ได้นำมาประยุกต์ใช้ ใช้สร้างและคำนวณการเคลื่อนที่ตาม ค่ามุมมองของแต่ละแกน โดยใช้คำสั่งดังนี้

คำสั่งที่ใช้ภายใน Module CalculPosition

3.3.1.1 glGetDouble เป็นคำสั่งในการรับข้อมูลที่เป็นหน่วย double คือ 64-bit floating-point เพื่อเก็บมาไว้เป็นตัวแปร

3.3.1.2 glModelViewMatrix คำสั่งในการนำข้อมูลที่ได้นำมาสร้างชุด Matrix เพื่อใช้ในการสร้างภาพและการเคลื่อนไหวของภาพ โดย จะได้รับค่าการคำนวณจาก Module CalculPosition

3.4 Display แสดงผลการทำงานของโปรแกรม

การแสดงผลการทำงานของโปรแกรม แบ่งส่วนการแสดงผลได้ ดังนี้

3.4.1 ส่วนของการป้อนข้อมูลเพื่อควบคุมการจำลองการเคลื่อนที่

3.4.1.1 Control Manual

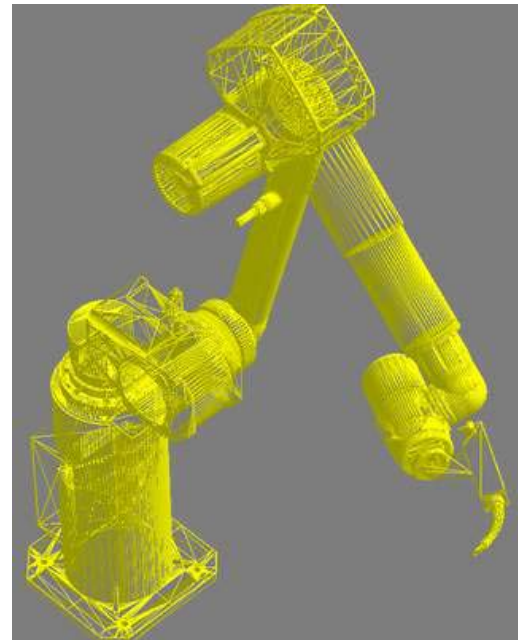
ป้อนข้อมูลเป็นตำแหน่งองศา (Joint Space) ที่ต้องการให้ แต่ละ Joint มีการเคลื่อนที่ โดยแต่ละ Joint มีการกำหนด มุมการเคลื่อนที่ สูงสุดและต่ำสุดอยู่แล้ว หรือสามารถเลื่อน Slider เพื่อเป็นการป้อนข้อมูลได้

3.4.2 ส่วนของการเลือกรูปแบบการแสดงผลกราฟฟิค

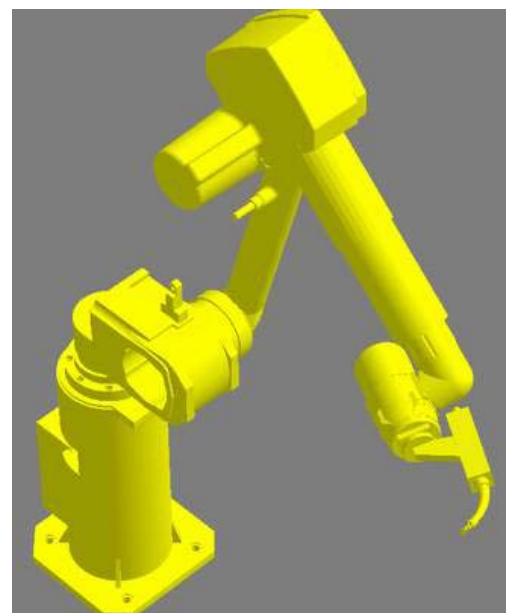
เป็นการเลือกรูปแบบการแสดงผลพื้นผิวของ แต่ละ Link โดยจะแบ่งการแสดงผลออกเป็น 3 แบบ และมีชนิดของพื้นผิว ดังนี้

3.4.2.1 Mesh แสดงพื้นผิวแบบโครงตาข่ายสามเหลี่ยม ตามข้อมูลของไฟล์ *.STL

3.4.2.2 Shadow แสดงพื้นผิวแบบ รูปทรงตัน พื้นผิวเป็นสีเหลือง

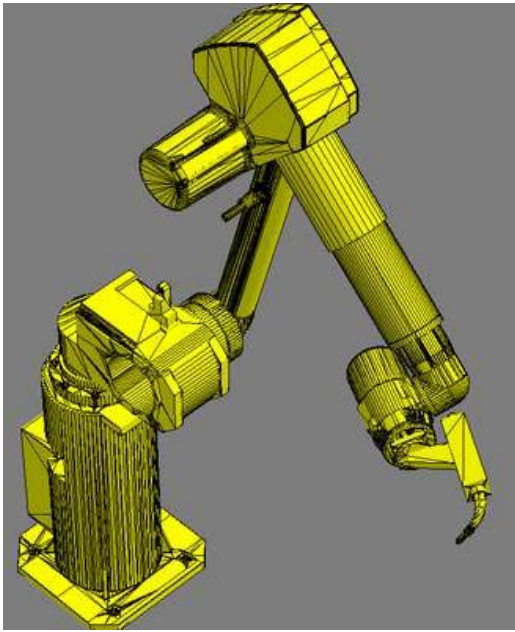


รูปที่ 6 แสดงภาพพื้นผิวแบบ Mesh



รูปที่ 7 แสดงภาพพื้นผิวแบบ Shadow

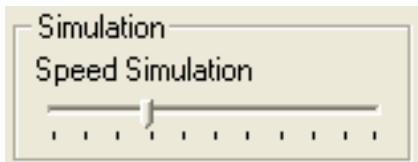
AMM089



รูปที่ 8 แสดงภาพพื้นผิวแบบMesh กับ Shadow

3.4.3 Speed Simulation

เป็นการกำหนดความเร็วของการเคลื่อนที่ของRobot โดยการใ้ การเลื่อน Slider เป็นตัวกำหนดความเร็ว การเคลื่อนที่ที่เร็วขึ้นถ้า เลื่อน Slider มาทางซ้ายสุด

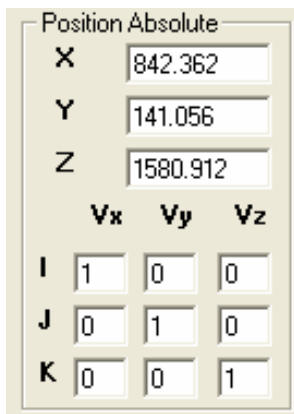


รูปที่ 9 Slider ปรับความเร็วการเคลื่อนที่

3.4.4 การแสดงค่าตำแหน่งของการเคลื่อนที่และข้อมูลชุดคำสั่งเคลื่อนที่

3.4.4.1 Position Absolute

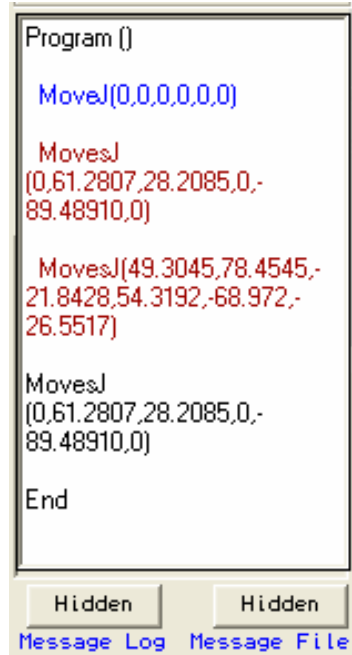
แสดงค่าตำแหน่ง X, Y, Z จากจุดต่อโยง Joint สุดท้าย เทียบกับ Origin



รูปที่ 10 แสดงค่าตำแหน่ง X, Y, Z Position Absolute

3.4.5 Message File

แสดงข้อมูลชุดคำสั่งควบคุมการเคลื่อนที่ สามารถแก้ไขหรือสร้าง ชุดข้อมูลใหม่ได้ มีรูปแบบข้อมูลควบคุมแบบ Joint Space ดังตัวอย่าง ด้านล่าง MovesJ(0,61.2807,28.2085,0,-89.48910,0) อธิบายได้ดังนี้ มีการเคลื่อนที่แบบ Joint Space Joint 1, 4, 6 ไม่มีการเคลื่อนที่ Joint 2 มีการเคลื่อนที่ทวนเข็มนาฬิกา 61.2807 องศา Joint 3 มีการเคลื่อนที่ทวนเข็มนาฬิกา 28.2085 องศา Joint 5 มีการเคลื่อนที่ตาม เข็มนาฬิกา 89.48910องศา



รูปที่ 11 แสดงข้อมูลชุดคำสั่งควบคุมการเคลื่อนที่

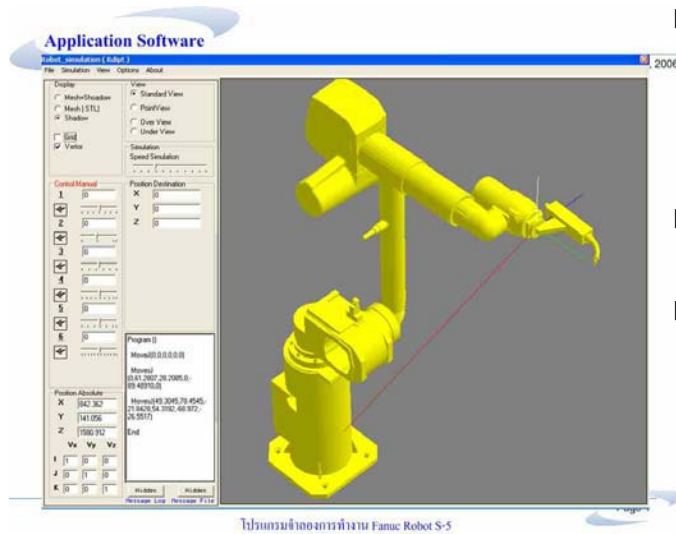
4. ส่วนของการเก็บข้อมูลการเคลื่อนที่

จะเป็นการเก็บข้อมูลของการเคลื่อนที่เริ่มตั้งแต่ข้อมูลองศาที่ป้อน ให้จนถึงพิกัดของหุ่นยนต์ และแสดงเป็นตารางบอกเป็นค่าตัวเลข สามารถนำข้อมูลที่ได้มานำมาสร้างชุดคำสั่งในการสั่งงานการเคลื่อนที่ ได้ แสดงดังตาราง

ตารางที่ 3 แสดงค่ามุมมองการเคลื่อนที่และพิกัดที่เก็บค่าได้

Step	Axis	Axis	Axis	Axis	Axis	Axis	X	Y	Z
	1	2	3	4	5	6			
1	45	20	50	10	90	90	522.	695.	972.
							929	407	981
2	45	20	50	10	90	0	522.	695.	972.
							929	407	981
3	45	20	50	10	0	0	468.	666.	844.
							862	143	053
4	0	-20	10	90	0	0	536.	38.5	1748.
							335	56	939
5	0	20	-65	90	0	0	746.	38.5	2039.
							018	56	379

5. สรุปผลการวิจัย



รูปที่ 12 โปรแกรมจำลองการทำงานของหุ่นยนต์ Fanuc Robot S-5

โครงการนี้เป็นการใช้คอมพิวเตอร์กราฟิกจำลองการทำงานของหุ่นยนต์เคลื่อนที่อิสระ 6 แกน Fanuc Robot-S5 เพื่อศึกษาการทำงานกลไกการเคลื่อนที่ของหุ่นยนต์ โดยโปรแกรมนี้จะช่วยลดความเสี่ยงการเกิดความเสียหายอันจะเกิดจากการปฏิบัติงานจริง

โดยโปรแกรมถูกพัฒนาด้วย Microsoft Visual Basic 6 เริ่มจากการนำไฟล์ *.STL ซึ่งเก็บข้อมูลพื้นที่ผิวของชิ้นงานเข้ามาสู่ไลบรารี OpenGL, GLU และ GLUT ช่วยในการเขียน และแสดงภาพกราฟิก

ตัวโปรแกรมมีการติดต่อกับผู้ใช้ที่ง่ายต่อการใช้งาน มีเครื่องมือต่าง ๆ เช่น การเปลี่ยนมุมมองของภาพ การป้อนข้อมูล การแสดงผลข้อมูล การเก็บข้อมูล (Data Logger) เป็นต้น โปรแกรมยังสามารถรายงานความผิดพลาดของการเคลื่อนที่ได้ขณะทำการแสดง

โปรแกรมสามารถแสดงให้เห็นการทำงานของหุ่นยนต์เคลื่อนที่อิสระ 6 แกน Fanuc Robot-S5 และทำงานได้ตามวัตถุประสงค์ที่ตั้งไว้ ทั้งยังสามารถนำไปพัฒนาให้โปรแกรมนี้มีส่วนเชื่อมต่อกับชุดควบคุมการเคลื่อนที่ของหุ่นยนต์จริงโดยผ่านทางพอร์ตที่ใช้รับส่งข้อมูล เช่น พอร์ตอนุกรม หรือพอร์ตขนาน ทำให้ผู้ใช้สามารถควบคุมหุ่นยนต์ได้โดยตรง และทดสอบการเคลื่อนที่จากโปรแกรมก่อนปฏิบัติงานจริง จะช่วยทำให้เกิดความปลอดภัย อีกทั้งยังช่วยประหยัดเวลาและค่าใช้จ่ายมากกว่าการใช้หุ่นยนต์จริงในการทดสอบ

6. กิตติกรรมประกาศ

ขอขอบคุณศูนย์เชี่ยวชาญเฉพาะทางแม่พิมพ์ยาง สถาบันค้นคว้าและพัฒนาเทคโนโลยีการผลิตทางอุตสาหกรรม ที่ให้การสนับสนุนส่วนหนึ่งในโครงการนี้

เอกสารอ้างอิง

[1] Edward Angel. OpenGLTM : A Primer. Addison-Wesley, Inc. 2002.

อ้างอิงส่วนของคำสั่งที่ใช้ในการเคลื่อนที่และการสร้างภาพ

[2] Mark J. Kilgard, The OpenGL Graphics Utility Toolkit (GLUT) Programming Interface API Version 3. Siligon Graphics, Inc. 1996 อ้างอิงส่วนของคำสั่งที่ใช้ในการเคลื่อนที่และการสร้างภาพ

[3] OpenGL [http:// www.opengl.org](http://www.opengl.org) (accessed on January 2006).

[4] OpenGL <http://rocbo.lautre.net/spip/> (accessed on February 2006).