บทที่ 4

ผลการดำเนินการวิจัย

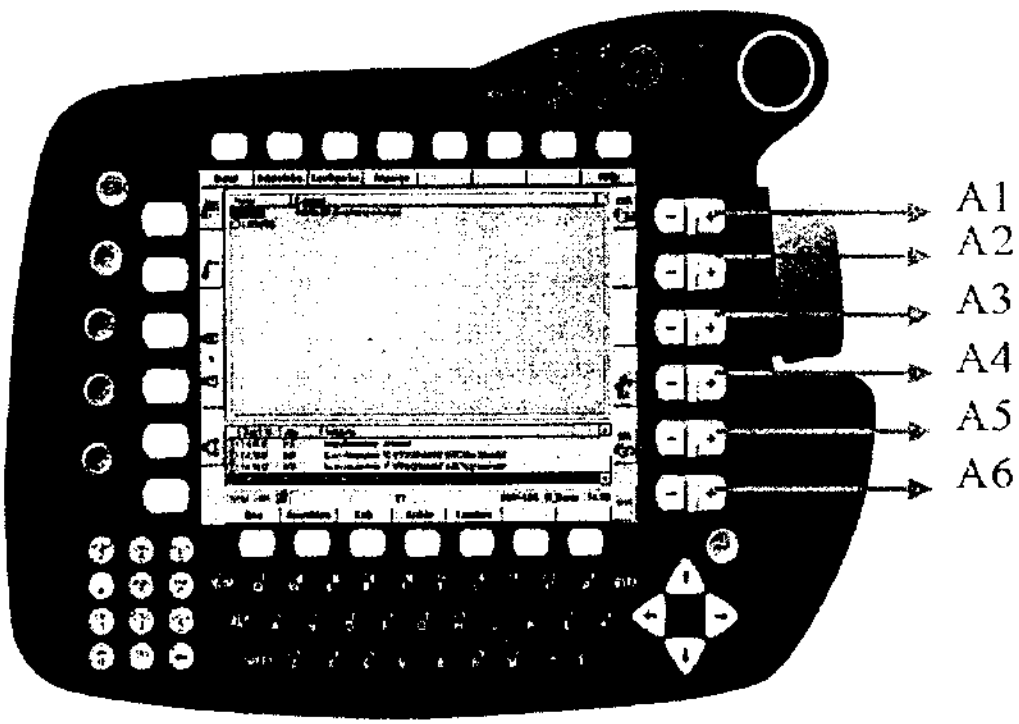
4.1 ศึกษาระบบการเคลื่อนที่และระบบควบคุมของหุ่นยนต์อุตสาหกรรม

KUKA Robot - KR125

1.ระบบควบคุมการเคลื่อนที่ของหุ่นยนต์อุตสาหกรรม KUKA Robot - KR125

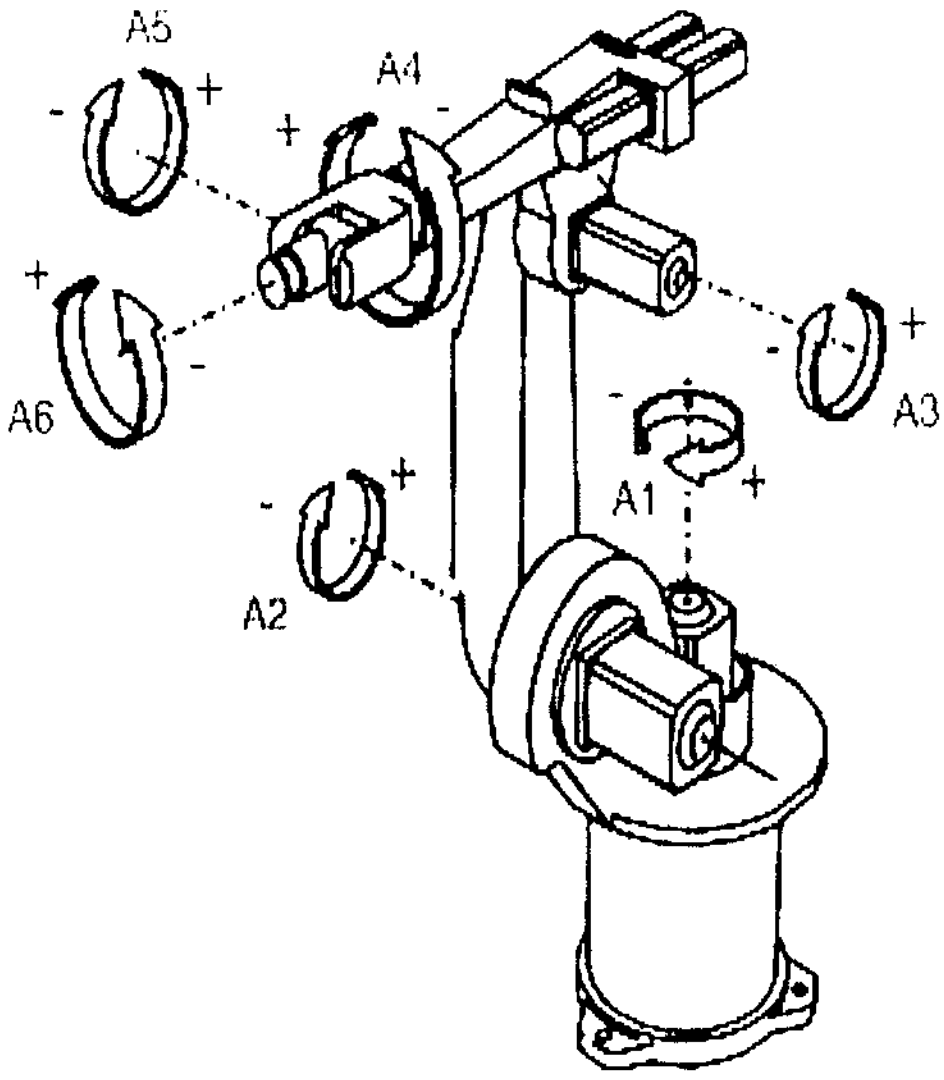
-ในการกำหนดทิศทาง + และ – ของแต่ละแกนตั้งแต่แกนที่ 1 ถึง 6 ของหุ่นยนต์ในระบบ Joint Coordinate โดยสามารถทำได้โดย

1. การกด Safety Switch ด้านหลัง Teach Pendant
2. กดเลือกระบบการเคลื่อนที่ให้เป็นระบบ Joint Coordinate
3. ทดลองกด + และ – ของแกนที่ 1 ถึง 6



รูปที่ 4.1 แสดง Key บน Teach Pendant ที่ใช้งานในการเคลื่อนที่หุ่นยนต์

หลังจากทดลองเคลื่อนที่ในแต่ละแกนแล้วได้ผลการเคลื่อนที่ในทิศทาง + และ – แสดงใน รูปที่ 4.2



รูปที่ 4.2 แสดงผลการทดลองการเคลื่อนที่ในทิศทาง + และ – ตั้งแต่แกนที่ 1ถึง 6

2. ทดลองหาค่ามุมของการเคลื่อนที่ในแต่ละแกนว่าสามารถเคลื่อนที่ได้มากที่สุดเท่าใด

โดยการทดลองเคลื่อนที่ระบบ Joint Coordinate

-ในการกำหนดทิศทาง + และ – ของแต่ละแกนตั้งแต่แกนที่ 1 ถึง 6 ของหุ่นยนต์ในระบบ Joint Coordinate โดยสามารถทำได้โดย

1. การกด Safety Switch ด้านหลัง Teach Pendant

2. กดเลือกระบบการเคลื่อนที่ให้เป็นระบบ Joint Coordinate

3. กดเลือกการ Monitor / เลือก 1.Act position / เลือก 1. Joint เพื่อเป็นการแสดงมุมองศาการเคลื่อนที่

4. ทดลองกด + และ – ของแกนที่ 1 ถึง 6 โดยเคลื่อนที่ในแต่ละแกน แต่ละทิศทาง จนกระทั่งหุ่นยนต์ไม่สามารถเคลื่อนที่ไปได้แล้วในแกนนั้นๆ

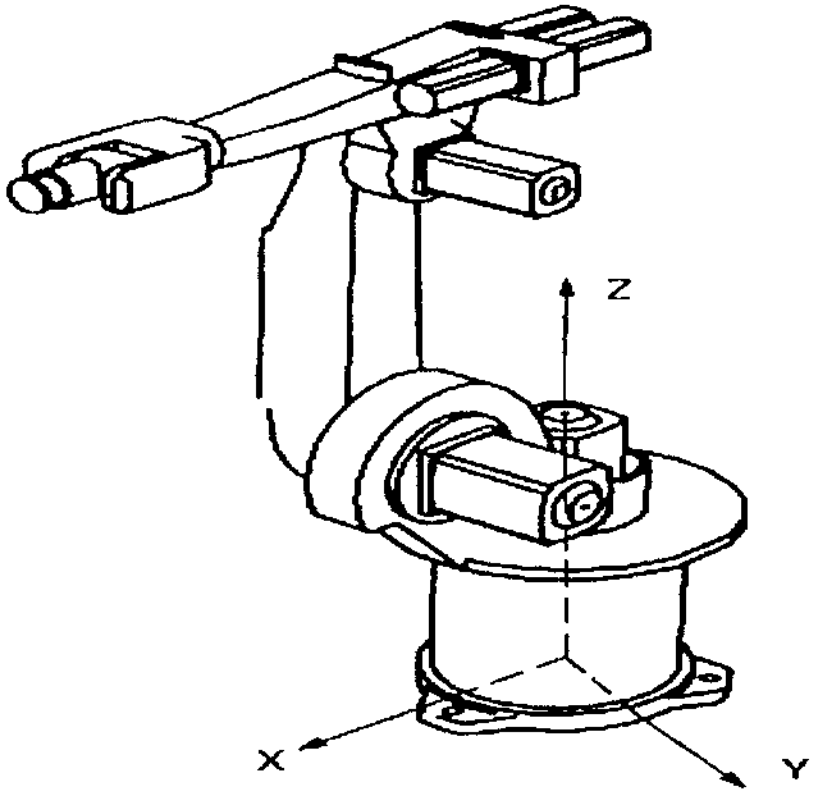
***** ในการทดลองการเคลื่อนที่ในแต่ละแกน ต้องระวังแกนอื่นๆที่ไม่ได้เคลื่อนที่ด้วย ต้องทำให้แกนที่ไม่ได้เคลื่อนที่ไม่กระทบกับแกนที่ทดลองเคลื่อนที่อยู่ ******

หลังจากทดลองเคลื่อนที่แต่ละแกนแล้วได้ผลการทดลองดังนี้

ตารางที่ 4.1 แสดงค่ามุมองศาที่เคลื่อนที่ได้ในทิศทาง + และ – ทั้ง 6แกน

แกนที่	มุมองศาที่เคลื่อนที่ได้ในทิศทาง -	มุมองศาที่เคลื่อนที่ได้ในทิศทาง +
แกนที่ 1	-185	185
แกนที่ 2	-130	3
แกนที่ 3	-120	148
แกนที่ 4	-350	350
แกนที่ 5	-120	120
แกนที่ 6	-350	350

3.ทดลองหาค่าพิกัดของปลาย Tool ที่ตำแหน่ง Home



รูปที่ 4.3 แสดงตำแหน่งพิกัด 0,0,0 ของหุ่นยนต์อุตสาหกรรม

จากการเคลื่อนที่ไปยังตำแหน่ง Home และ Monitor ค่าพิกัดจะได้ดังนี้

1.การ Monitor ในระบบ Cartesian

ตารางที่ 4.2 แสดงค่าที่ Monitor ในระบบ Cartesian

Position < mm >	Orientation < deg >
X = 1765.187	A = 2.054
Y = -9.491	B = 2.372
Z = 1645.614	C = 179.427

2.การ Monitor ในระบบ Joint

ตารางที่ 4.3 แสดงค่าที่ Monitor ในระบบ Joint

Axis Angle < deg >	Axis Angle < deg >
A1 = 0.271	A4 = 1.341
A2 = -88.881	A5 = 0.526
A3 = 91.129	A6 = -1.943

3.การ Monitor ในระบบ Incremental

ตารางที่ 4.4 แสดงค่าที่ Monitor ในระบบ Incremental

Incremental < Incr >	Incremental < Incr >
I1 = 2275	I4 = 5345
I2 = -744292	I5 = 1896
I3 = 767272	I6 = -4777

4.2 ศึกษาระบบควบคุมและสั่งงานหุ่นยนต์อุตสาหกรรม KUKA Robot - KR125

ทดลองควบคุมหุ่นยนต์อุตสาหกรรมโดยการป้อนคำสั่งผ่านทางโปรแกรม Text Document

1.ทำการวัดพิกัดจากตำแหน่งฐานของRobot โดยการบังคับด้วย Tech Pandant และขณะเดียวกันทำการ Monitorในระบบ Cartesian ด้วย บังคับไปที่จุดต่างๆของขอบโต๊ะชิ้นงานที่นำมาใช้ประกอบการทดลอง โดยขนาดที่วัดจาก Monitorในระบบ Cartesian ได้ดังตารางที่ 4.5

ตารางที่ 4.5 แสดงขนาดที่วัดจาก Monitorในระบบ Cartesian

ตำแหน่งของโต๊ะชิ้นงาน	X(mm.)	Y(mm.)	Z(mm.)
Point1	2257.5	911	750
Point2	2257.5	-911	750
Point3	1500	-911	750
Point4	1500	911	750

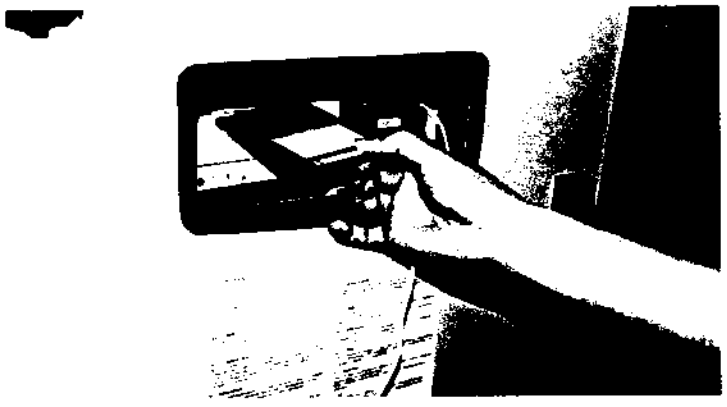
2.นำพิกัดที่ได้ทั้ง 4 จุดมาเขียนลงในโปรแกรม Text Document ซึ่งมีรายละเอียดดังนี้คือ

```
DEF AT001                                   ;กำหนดชื่อ file
DECL AXIS HOME                              ;ประกาศตัวแปร
BAS (#INIMOV,0)
HOME ={AXIS:A1 0,A2 -90,A3 90,A4 0,A5 0,A6 0}   ;กำหนดตำแหน่งเริ่มต้น(Home)
PTP HOME                                    ;เคลื่อนที่ไปตำแหน่ง Home
PTP {X 2257.5, Y 911,Z 850 }                ; เคลื่อนที่ไปตำแหน่ง safety point
LIN {X 2257.5, Y 911,Z 750}                 ; เคลื่อนที่ไปตำแหน่ง Point 1
LIN {X 2257.5, Y -911,Z 750}                ; เคลื่อนที่ไปตำแหน่ง Point 2
LIN {X 1500, Y -911,Z 750}                  ; เคลื่อนที่ไปตำแหน่ง Point 3
LIN {X 1500, Y 911,Z 750}                   ; เคลื่อนที่ไปตำแหน่ง Point 4
LIN {X 2257.5, Y 911,Z 750}                 ; เคลื่อนที่ไปตำแหน่ง Point 1
LIN {X 2257.5, Y 911,Z 850}                 ; เคลื่อนที่ไปตำแหน่ง safety point
PTP HOME                                    ;เคลื่อนที่ไปตำแหน่ง Home
END                                         ;จบการทำงาน
```

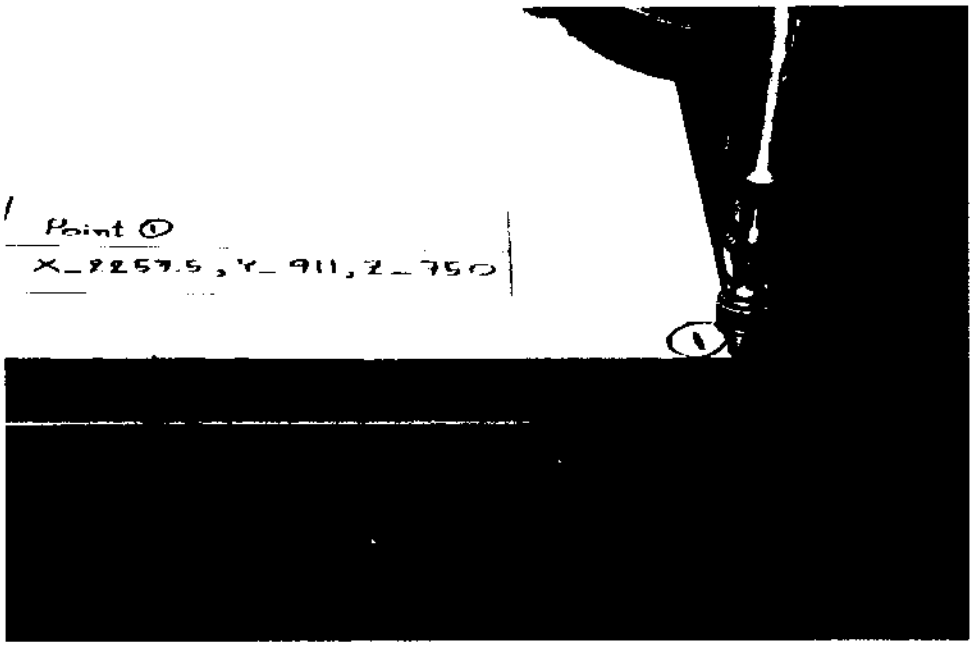
และบันทึกไฟล์ที่เขียนให้มีรูปแบบดังนี้

- File name = *.SRC
- Save as type = All Files [*.*]

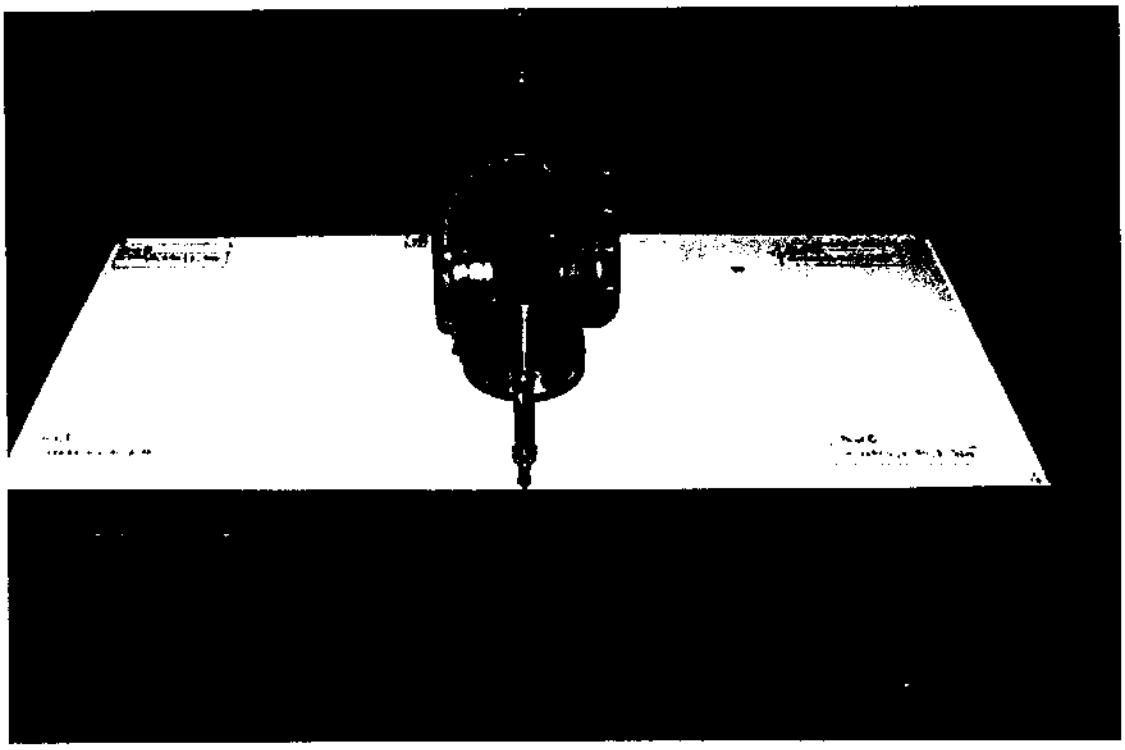
3.นำโปรแกรมที่ได้ป้อนเข้าไปใน Robot จะได้ผลการทดสอบดังรูปที่ 4.4 - 4.8



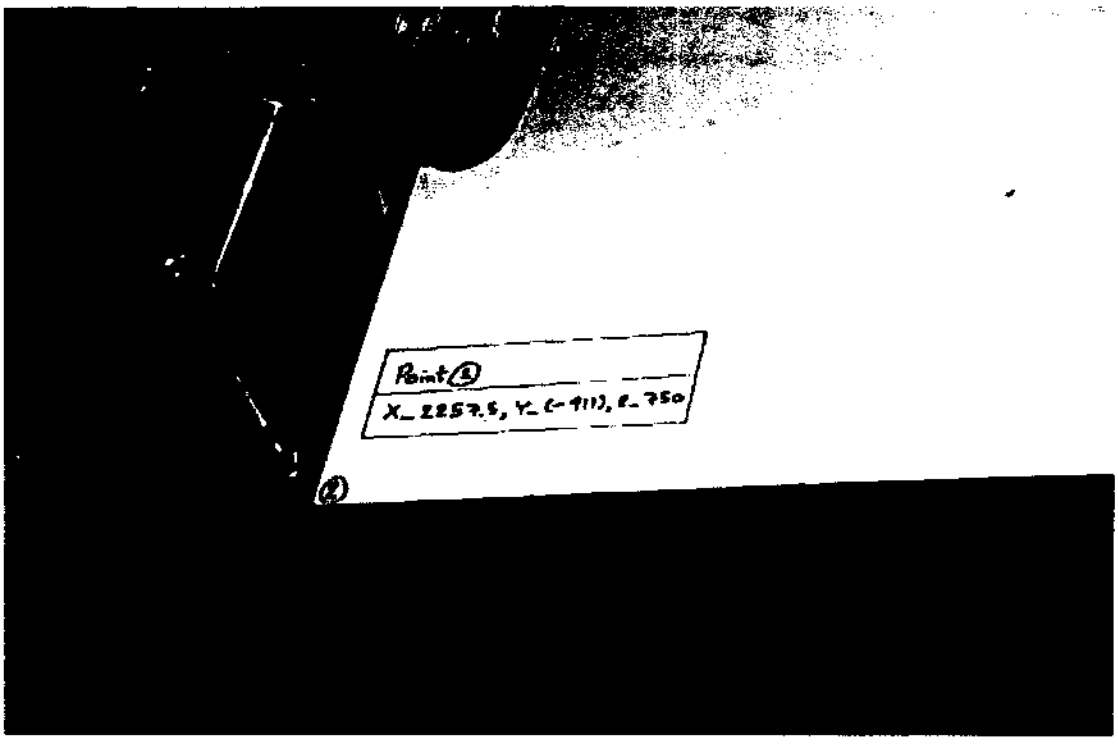
รูปที่ 4.4 แสดงการนำแผ่นบันทึกโปรแกรมได้ใส่เข้าไปใน Robot Control



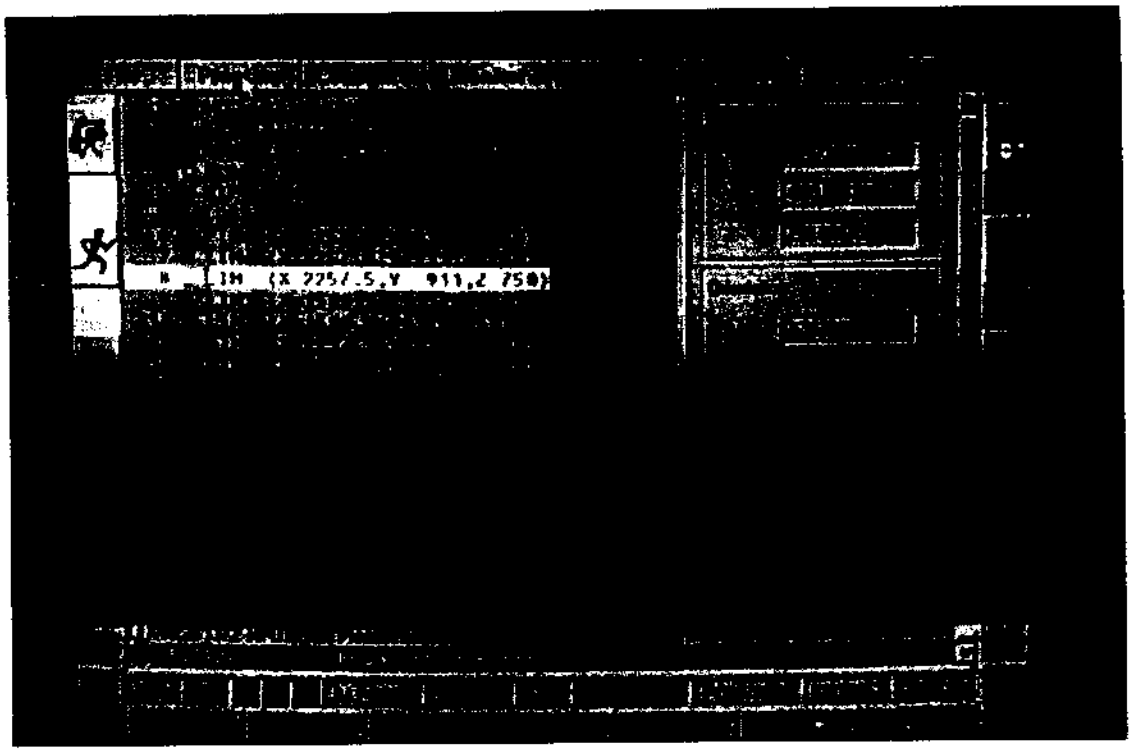
รูปที่ 4.5 แสดงการเคลื่อนที่ปลาย tool ของ Robot มาที่ safety point



รูปที่ 4.6 แสดงการเคลื่อนที่ปลาย tool ของ Robot มาที่ Point 1



รูปที่ 4.7 แสดงการเคลื่อนที่ปลาย tool ของ Robot ไปยัง Point 2

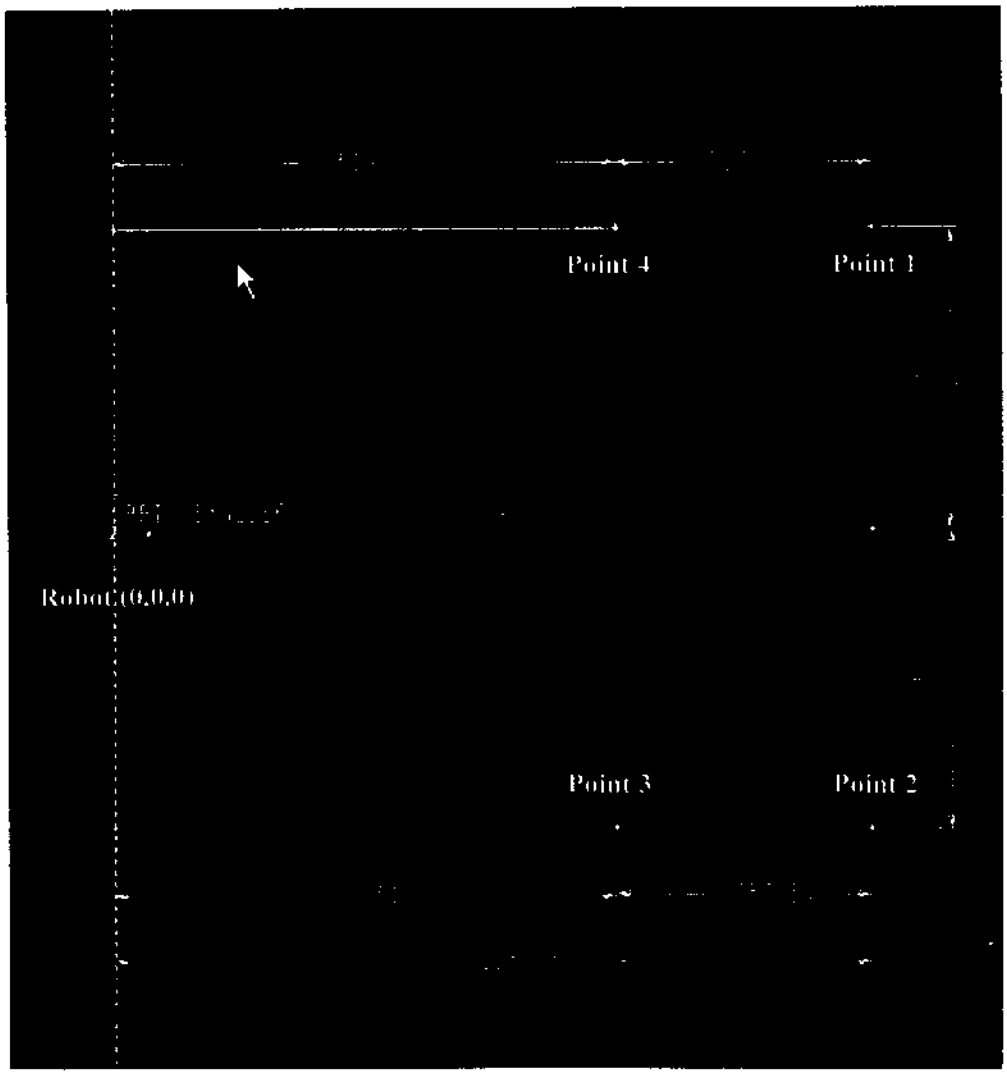


รูปที่ 4.8 แสดงหน้าจอของโปรแกรมควบคุมหุ่นยนต์อุตสาหกรรม

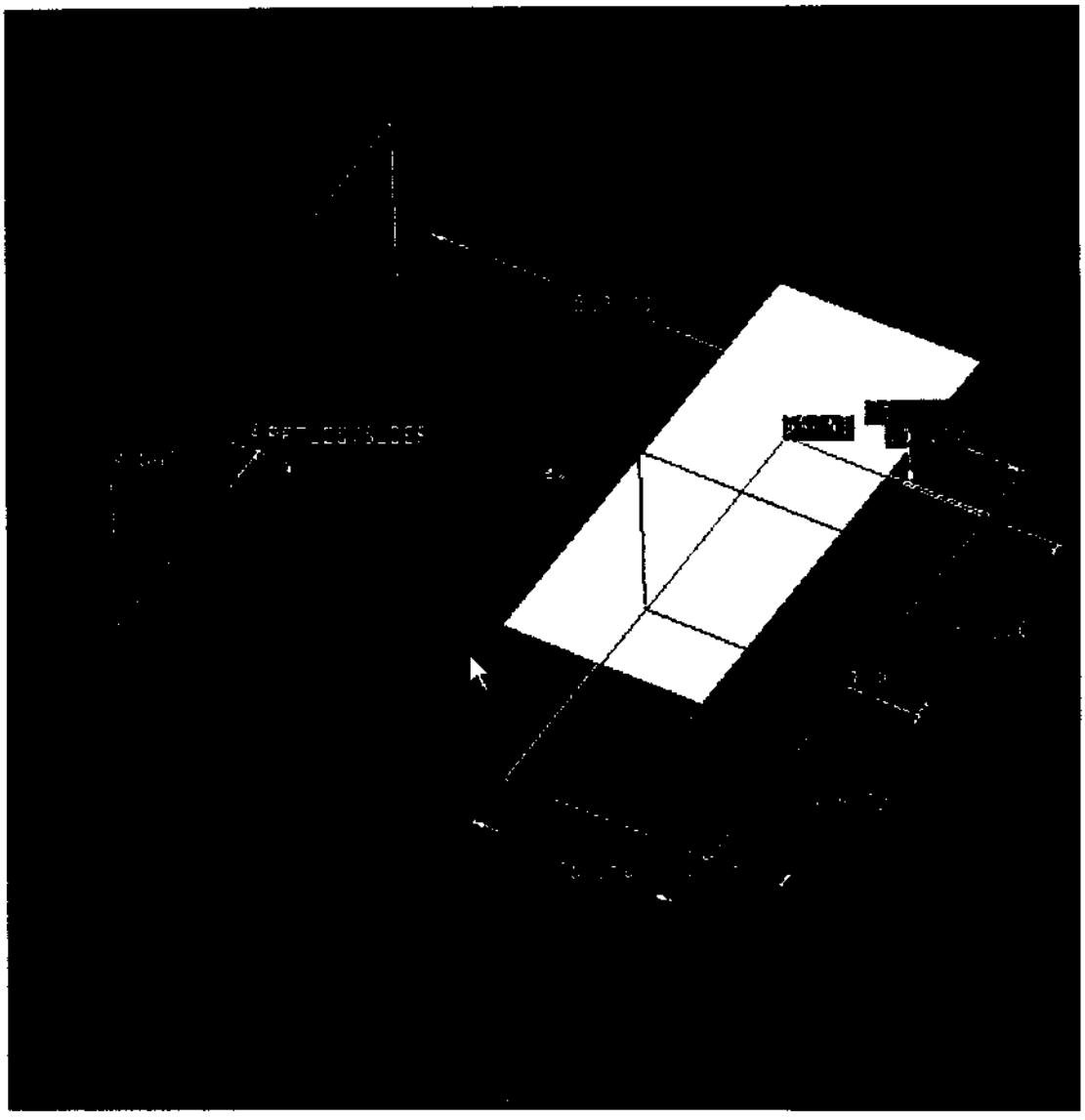
4.3 ศึกษาการใช้โปรแกรมช่วยในการเขียนแบบและช่วยในการผลิต(CAD/CAM)

การใช้โปรแกรม Proengineer 2000i^2 ช่วยในการเขียนแบบ คำนวณพิกัดจุดของชิ้นงาน และจำลองเส้นทางเดินของเครื่องมือ (Tool - path Simulation) โดยมีขั้นตอนดังนี้

1.นำข้อมูลพิกัดของโต๊ะชิ้นงานมาเขียนลงในโปรแกรม Proengineer 2000i^{2}ดังแสดงในรูปที่ 4.9 – 4.10



รูปที่ 4.9 แสดงระยะต่างๆของพื้นโต๊ะชิ้นงานเมื่อวัดจากฐาน Robot

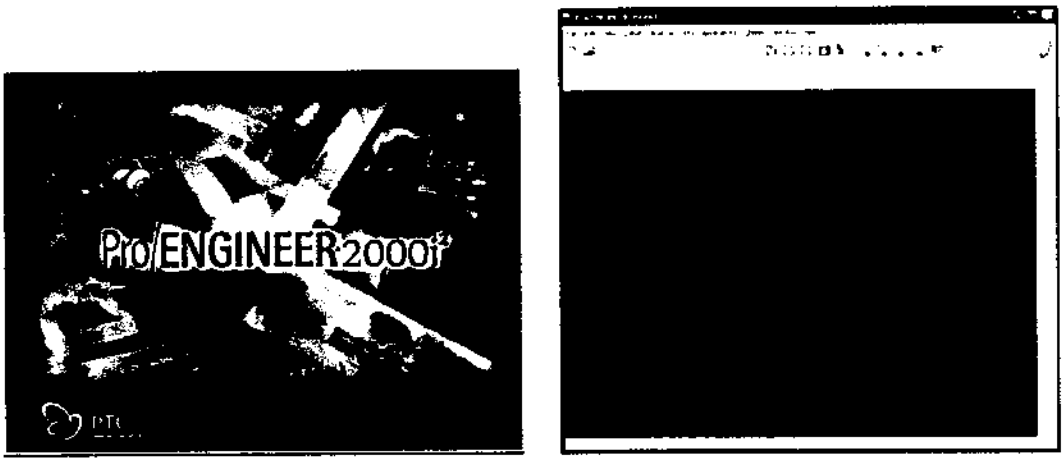


รูปที่ 4.10 แสดงตำแหน่งและขนาดของโต๊ะชิ้นงานเมื่อเทียบกับจุดเริ่มต้นของหุ่นยนต์

การสร้าง CAD มีขั้นตอนดังนี้

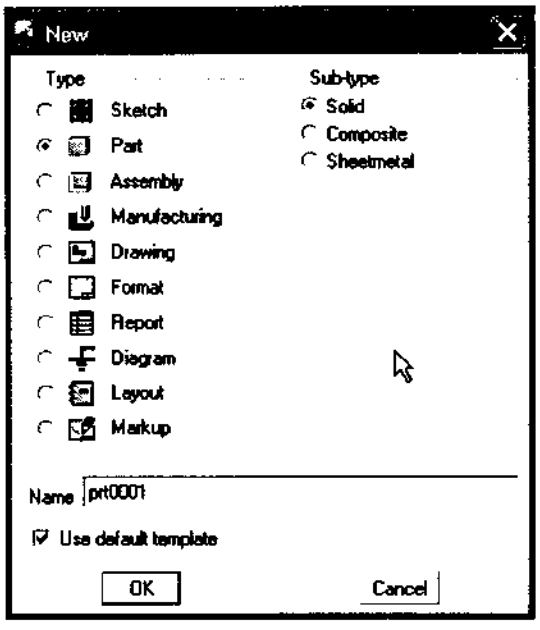
การวาดโต๊ะ

1. นำข้อมูลพิกัดที่ได้ตั้งจุดต่างไว้ จาก Robot ไปยัง โต๊ะ
2. เปิดโปรแกรม **Proengineer 2000i**^{2}



รูปที่ 4.11 แสดงในส่วนของหน้าตาโปรแกรม Proengineer2000i^2

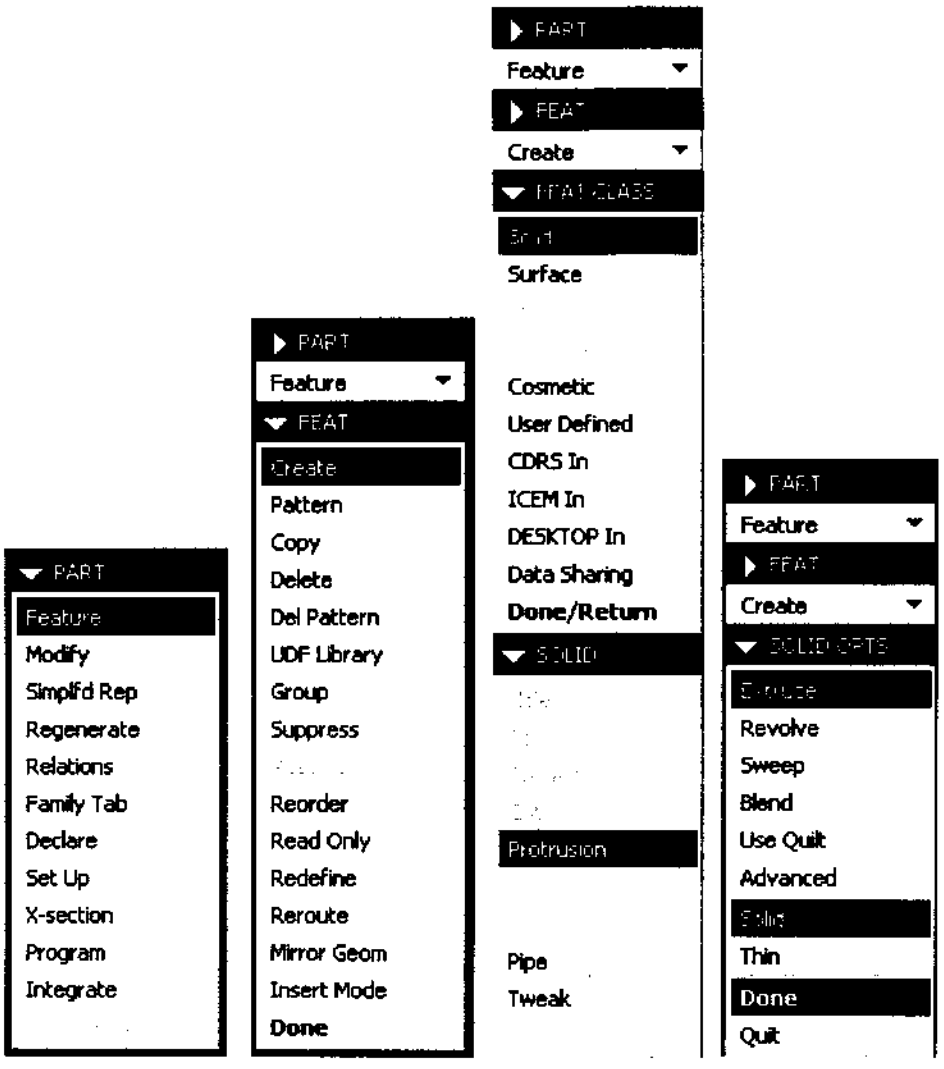
File>New / Part / Solid > Ok ใส่ชื่อ Part ที่ต้องการ



รูปที่ 4.12 การ Create New file

หมายเหตุ ขั้นตอนการ Set หน่วยและการวาดรูป 3 มิติ ดูได้จากภาคผนวก ก.
จากนั้นทำการ Create วาดรูปตามพิกัดโดยทำการเลือกจาก Pop up Menu

Feature / Create / Solid / Protrusion / Extrude / Solid / done / One Side / done

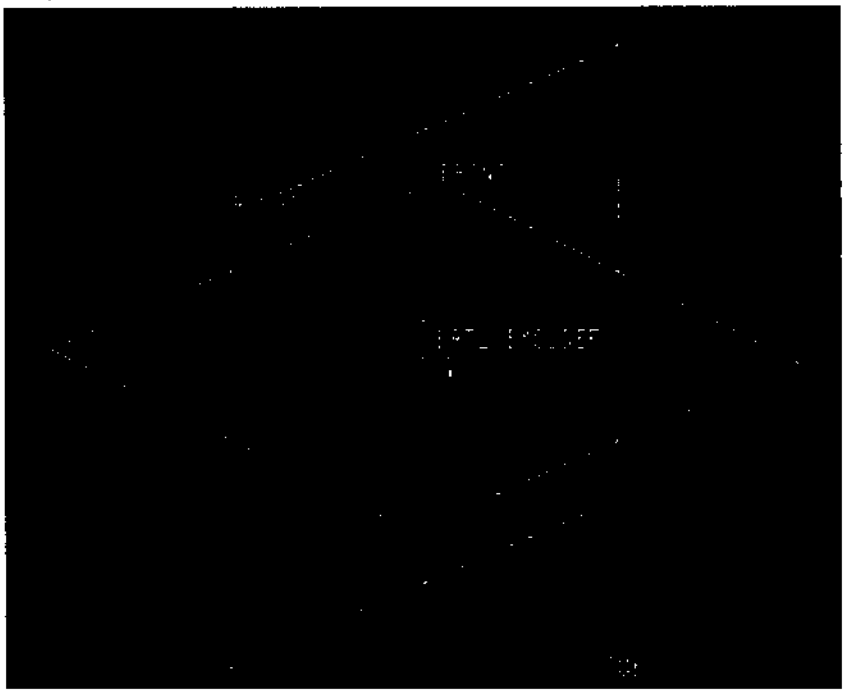


จะปรากฏรูปร่างลักษณะของ View และมุมมองของการเลือก View ในการสร้าง Plan

รูปที่ 4.13 แสดงขั้นตอนการเขียน Drawing

เลือก Plan โดยการหมุน Plan กด Ctrl + (mouse ปุ่มกลาง) ให้ได้ Plan ตามรูปนี้

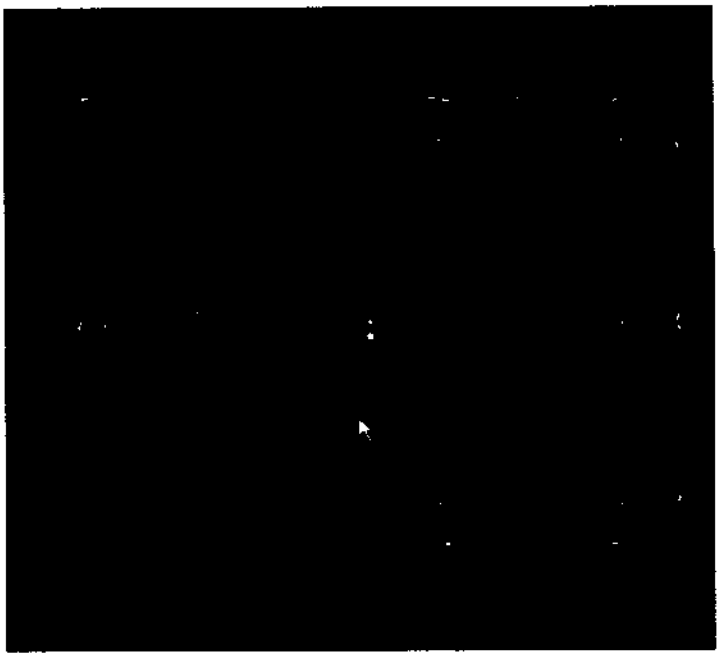
สีแดง คือ มุมมองของแกน X, สีเขียว คือมุมมองของแกน Y, สีฟ้า คือ มุมมองของแกน Z



รูปที่ 4.14 มุมมองการเลือก Plan

เลือก Plan Front / **Okay** แล้วก็ Top / Plan Top จะได้ view ในการสร้างชิ้นงาน

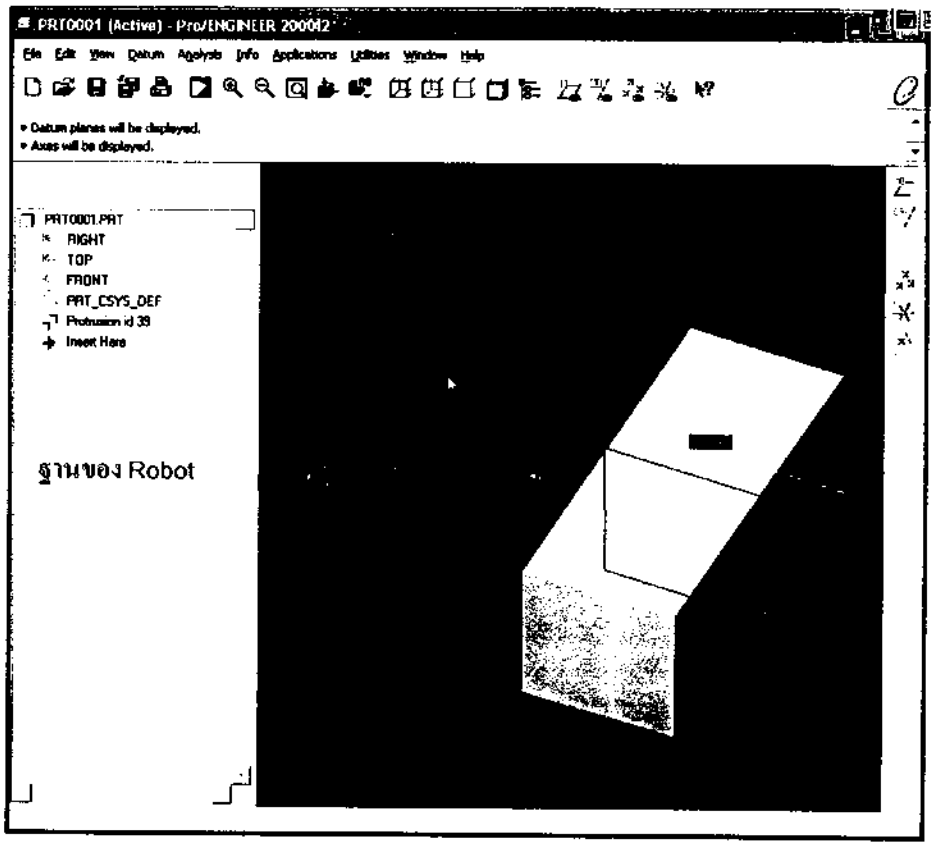
วาดรูปตามข้อมูลข้างต้นที่ได้โดย Click ที่ ↖ ด้านข้างของโปรแกรม



รูปที่ 4.15 แสดงการ Sketch

จากข้อมูล Point ทั้ง 4 จะได้รูปร่างดังรูปที่ 4.15

Click ถูกเพื่อที่จะใส่ความหนา ✔ ตามด้วย Blind => ใส่ความหนา 750 / **Preview / ok** จะได้รูปโดยสมมุติว่าเป็นโต๊ะที่จะใช้เป็นฐานในการวางชิ้นงานเพื่อให้ง่ายต่อระบบพิกัดในการเคลื่อนที่ของ Robot เป็นไปดังรูป



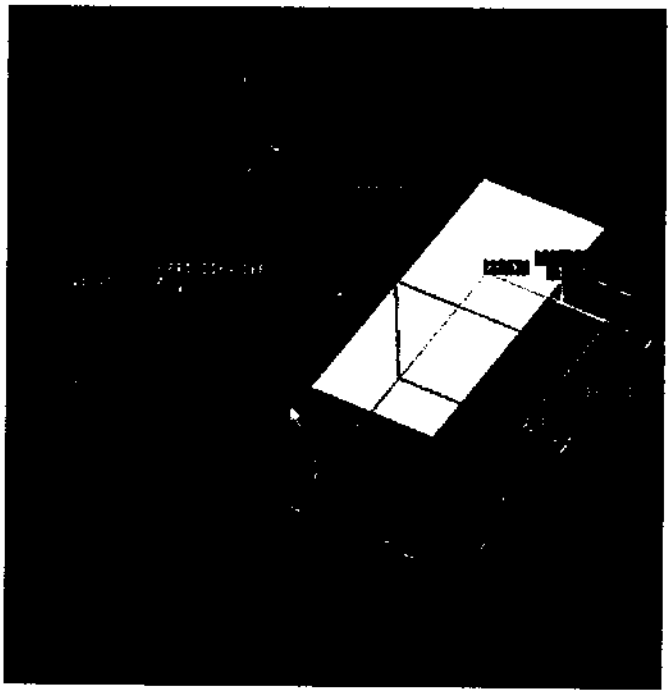
รูปที่ 4.16 ตำแหน่งของ Robot กับ โต๊ะ

ทำการบันทึกภาพที่ได้วาดขึ้นไปที่ File Save As ใส่ชื่อในช่อง New Name เป็น AN001 Click OK ปุ่ม OK

เมื่อได้พิกัดตำแหน่งของโต๊ะที่เกิดจากการวัดขนาดและระยะจากหุ่นยนต์นำมาสร้างเป็นรูปร่างได้แล้วตำแหน่งฐานของหุ่นยนต์สังเกตได้ว่าจะอยู่ห่างออกไปจากโต๊ะ ต่อไปจะเป็นการสร้างชิ้นงานก็จะเมื่อกับขั้นตอนการสร้างโต๊ะแต่ View ที่เลือกจะเป็นผิวของโต๊ะทางด้านบนด้านข้างทำให้ View ที่ออกมาอาจจะสับสนสักหน่อยและยากต่อการวาด

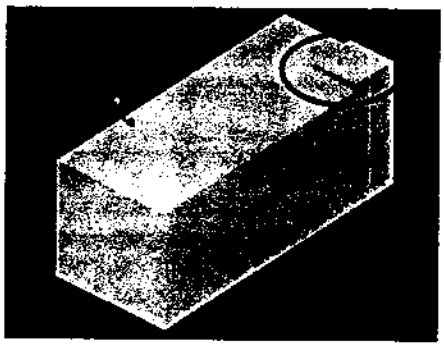
2. ออกแบบตัวอย่างชิ้นงานที่นำมาทดสอบการเคลื่อนที่ของหุ่นยนต์อุตสาหกรรมโดยมีขนาดและรูปร่างดังนี้

เริ่มจากการเรียก Flie ที่ได้สร้างโต๊ะไว้มาที่มีชื่อว่า AN001 โดยไปที่ Menu **File / Open**



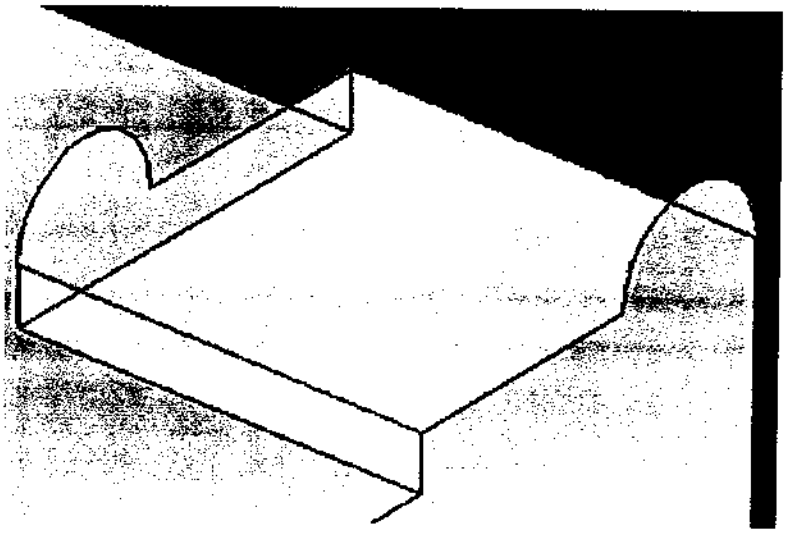
รูปที่ 4.17 พิกัดของโต๊ะ

ทำการสร้างชิ้นงานบนโต๊ะโดยไปที่ Menu Pop up เลือก **Feature / Create / Solid / Protrusion / Extrude / Solid / done / One Side / done** จะปรากฏรูปร่างลักษณะของ View และมุมมองของการเลือก View ในการสร้าง Plan เลือกที่ผิวด้านบน (**Flip** คือ เป็นการปรับเปลี่ยนหัวลูกศรว่าจะให้ชิ้นงานที่จะสร้างใหม่ขึ้นมาไปทางด้านไหน) Click **OK** เลือก default ก็จะได้ View เพื่อ Sketch ชิ้นงาน พอได้รูปร่างเสร็จแล้วจากนั้นทำการ Click ถูก เพื่อที่จะใส่ความหนา ✔ ตามด้วย Blind => ใส่ความหนา 40 / **Preview / ok**



รูปที่ 4.18 แสดงการขึ้นรูปสี่เหลี่ยม

ทำตามขั้นตอนเดิมที่กล่าวมาก็จะได้ชิ้นงานออกมาดังนี้

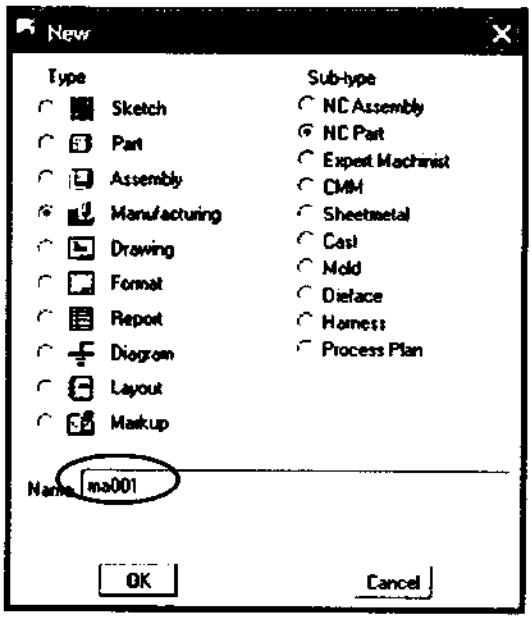


รูปที่ 4.19 แสดงขนาดและรูปร่างของตัวอย่างชิ้นงาน

Save As *.* file Name An002

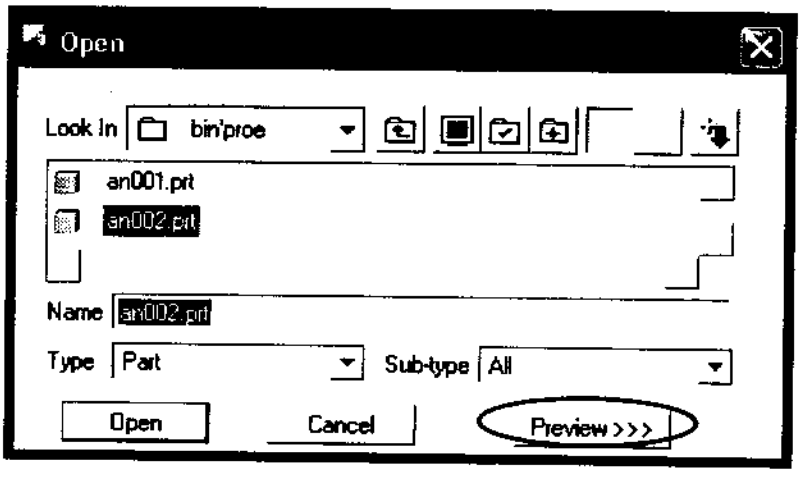
3. นำรูปภาพที่ได้สร้างขึ้นจากหัวข้อที่ผ่านมาทำการแปลงจาก CAD มาเป็น CAM โดยใช้การทำงานในส่วนของ Manufacturing เพื่อให้ได้ CLfile Data แต่ไม่ถึงกับ NC Post Processor ที่เป็น Code ของเครื่อง CNC เช่น เครื่องกัด เครื่องกลึง

3.1 เปิด Programs Proengineer 2000i2 จากนั้น Click ที่ Icon **New** หรือไปที่ **File / New** > เลือกในส่วนของ **Manufacturing / NC Part** >ตั้งชื่อ **Name : ma001** Click **OK** ดังรูป



รูปที่ 4.20 New file Manufacturing NC Part

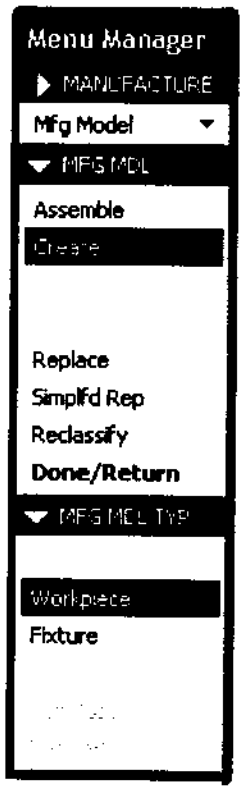
หาfile ชื่อ **An102** ถ้ามองไม่เห็นว่าใช่หรือไม่ให้ไป Click ที่ปุ่ม Button Preview >>>



รูปที่ 4.21 Open file เลือกชื่อ file

จากนั้นจึง Click Button **Open** Processing model AN002 ...จะ Load 100% has been completed. ที่ตำแหน่ง Message (แสดงว่ารูปที่เขียนขึ้นไม่มี error เกิดขึ้นทำให้รูปที่เขียนขึ้นสมบูรณ์สามารถสร้าง Manufacturing ต่อได้)

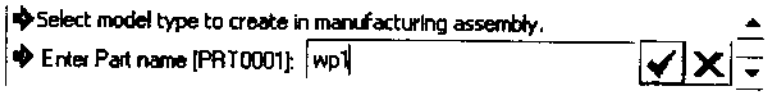
3.2 สร้าง Model เป็นชิ้นงานที่ต้องการเพื่อทำการ สร้างทางเดิน



(a)

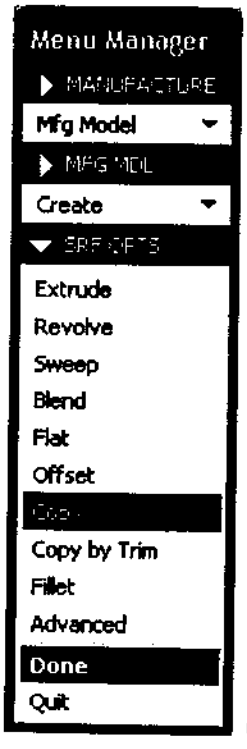
MFG Model > Create > Work piece

- ตั้งชื่อให้กับชิ้นงานในช่อง Enter part name (PRT001) : WP1



Click เครื่องหมาย ต่อไปจะเป็นการเลือกผิวชิ้นงาน

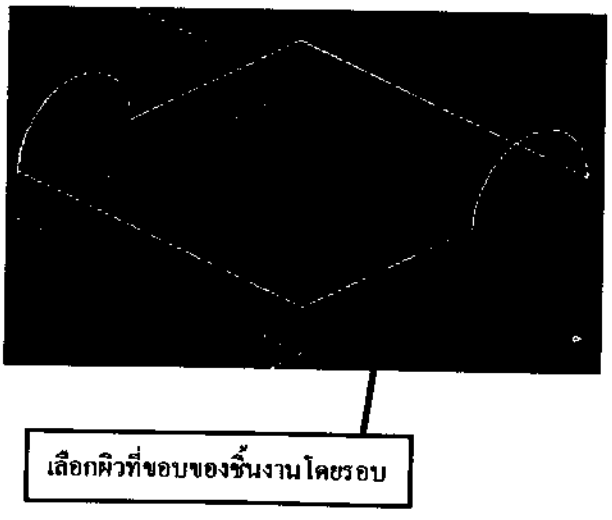
รูปที่ 4.22 Menu Popup สร้าง (a) , (b) Model , (c) แสดงการเลือกผิว



(b)

ที่ Pop up Menu >> **Surface / copy / Done**

<u>ข้อสังเกต</u> ผิวชิ้นงานที่ถูกเลือกจะเป็นสีเขียว



(c)

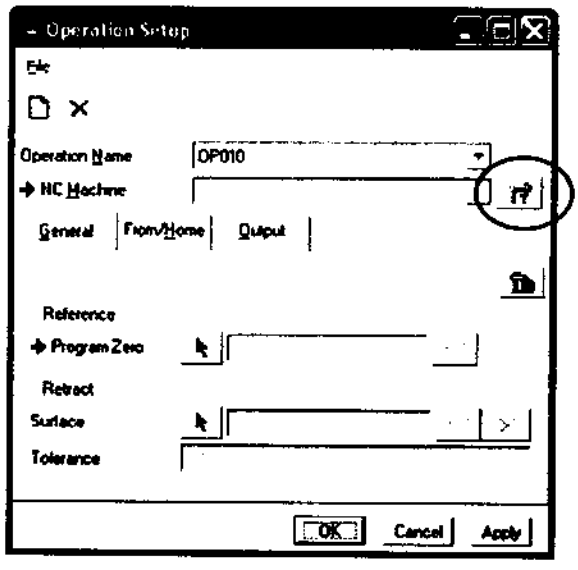
Include / Indiv Surfs / pick เป็นการเลือกผิวที่ต้องการสร้างทางเดินของTool

Done > OK

3.3 การกำหนดค่า Setup ของการใช้ระบบเครื่อง

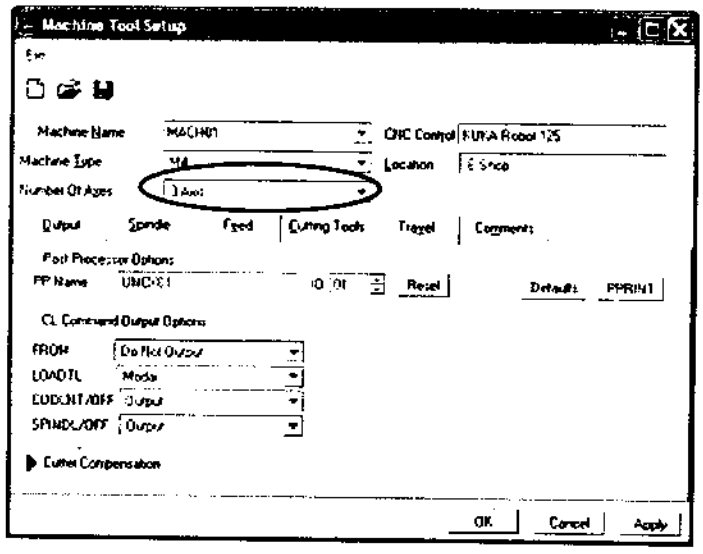
- โดยการใช้คำสั่ง Manufacturing Setup

Mfg Setup > จะปรากฏ **dialog box : Operation Setup**



เลือก Click ที่ช่อง NC Machine จะปรากฏ dialog box : Machine Tool Setup ทำการใส่ค่า

รูปที่ 4.23 Operation Setup



Click หัวลูกศรลง ในช่อง **Number Of Axes** สามารถกำหนดแกนได้ 3 Axis, 4 Axis, 5 Axis

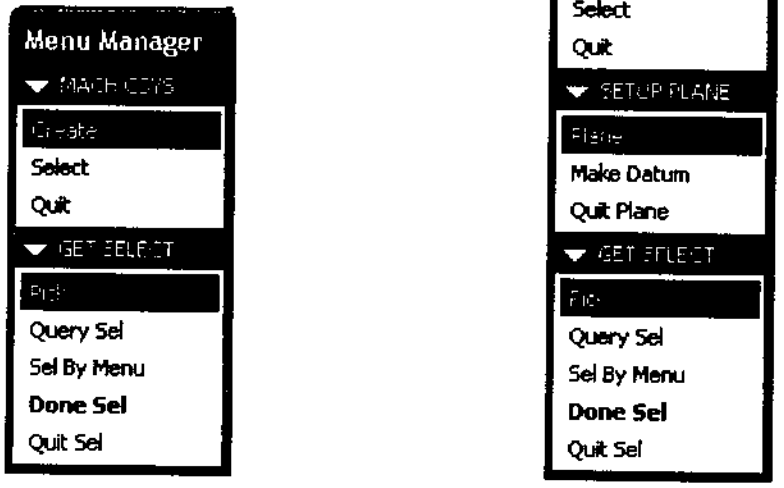
รูปที่ 4.24 Machine Tool Setup

กำหนดแกนให้เป็น 3 Axis Click **OK** (ค่าอื่นไม่ต้องไปเปลี่ยนเอาไว้ Set ที่หลังได้)

เลือกโคออดิเนตอ้างอิง โดย Click ที่ Reference → Program Zero จำเป็นต้องสร้าง โคออดิเนตขึ้นใหม่เพื่อป้องกันการสับสนกับระบบพิกัด xyz และเป็นการ Set ระบบพิกับของหุ่นยนต์อีกด้วยโดยเทียบกับการเคลื่อนที่แบบ World Coordinate system โดยมีวิธีการดังนี้

Create > เลือก โคออดิเนต CSI_CSYS_DEF หรือเลือกที่ชิ้นงานก็ได้

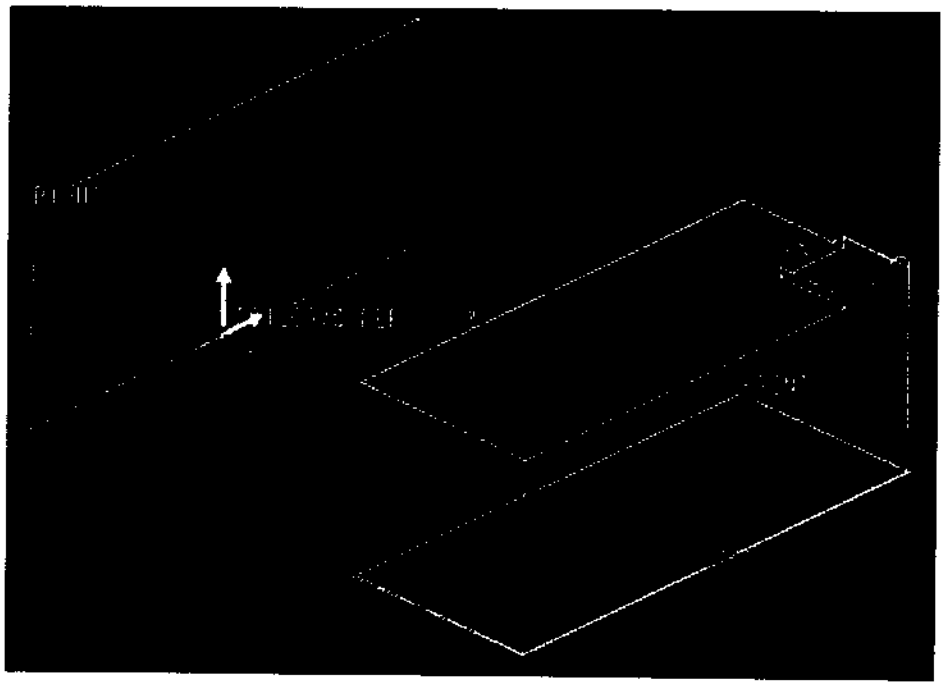
3 Planes / Done / เลือก Plane ทั้ง 3 (Top, Front, Right)



(a) (b)

รูปที่ 4.25 (a) , (b) แสดงการ Set โคออดิเนต

จะปรากฏลูกศรสีเหลืองแดงชี้ไปยังทิศทั้ง 3 สำหรับขั้นตอนในการกำหนดทิศทางของระบบ xyz โดยกำหนดให้เป็นไปตามระบบการเคลื่อนที่แบบ World Coordinate System



รูปที่ 4.26 รูปแสดงตำแหน่งของ Robot ในการ Set Coordinate กับชิ้นงาน

ที่ Pop up > **Next** คือ เปลี่ยนหัวลูกศรไปข้างหน้า โดยแถบสีแดงจะไปแทนที่สีเหลือง

Previous คือ เปลี่ยนหัวลูกศรไปข้างหน้า โดยแถบสีแดงจะไปแทนที่สีเหลือง

Reverse คือ เป็นการกลับด้านของหัวลูกศร

ถ้าได้ตำแหน่งแล้วก็ add ให้แกน โดย Next สีแดงอยู่แนวดิ่ง เลือกแกน Z จากนั้นจะขึ้นแกนพุ่งมาข้างหน้าให้เลือก X ที่เหลือจะเป็น Y

ต่อไปเป็นการสร้างรูปทรงเรขาคณิตเพื่อการ Machine

Mfg Geometry > Mill Surface > Create > ใส่ชื่อในช่อง Enter name for surface : SF1 กด Enter

Add > Copy / Done > Include / Indiv Surfs > เลือก Surface เดิมที่เคยเลือกมาแล้วข้างต้น

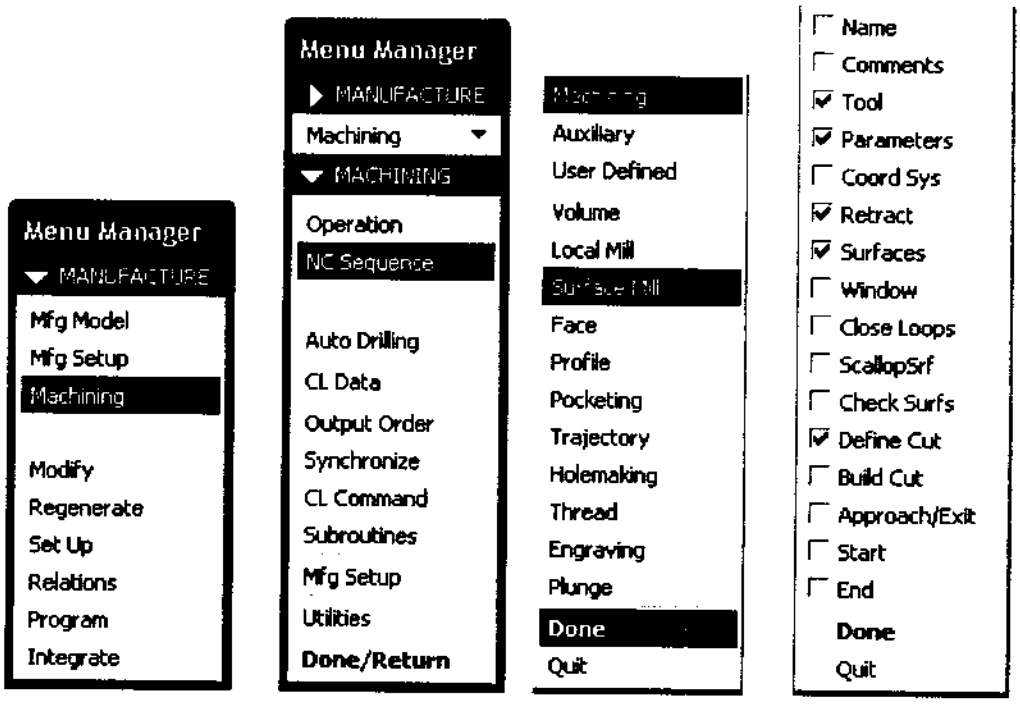
Click **OK**

3.3 การใช้คำสั่งในส่วน Machining

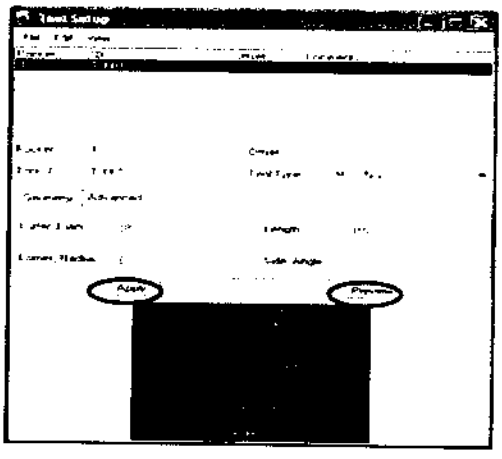
- เป็นขั้นตอนการกำหนดเครื่องมือ Tool และค่า Parameter ต่าง ๆ โดยทำดังนี้

Machining > NC Sequence > Surface Mill / Done

Click เครื่องหมายดังรูปจากนั้นกด Done



รูปที่ 4.27 (a) , (b) , (c) , (d) แสดงขั้นตอนการก่อนเข้า Set Tool

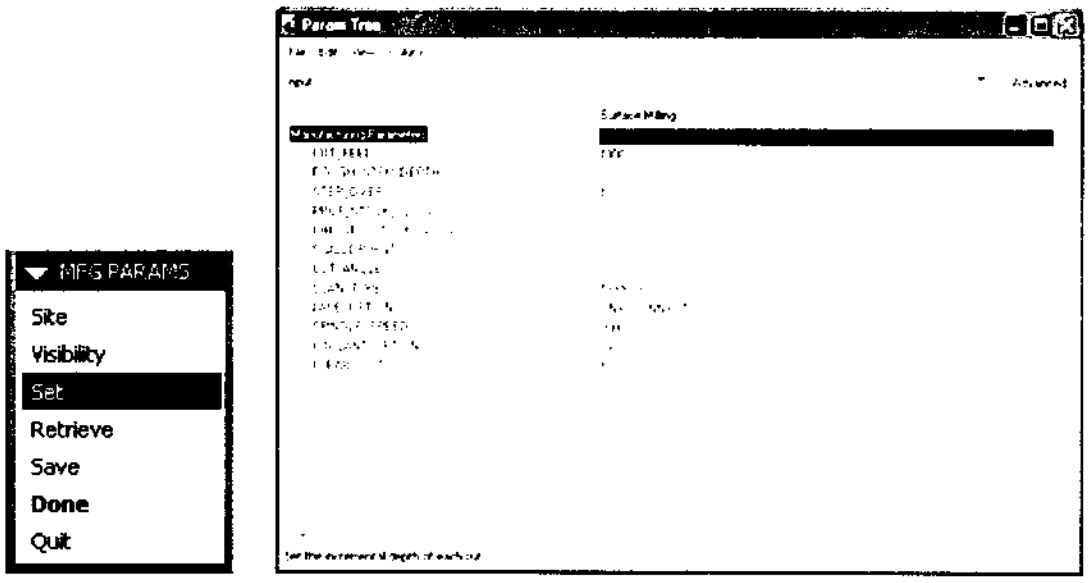


ใส่ขนาดของ Cutter_Diam: **8**

Length : **115**

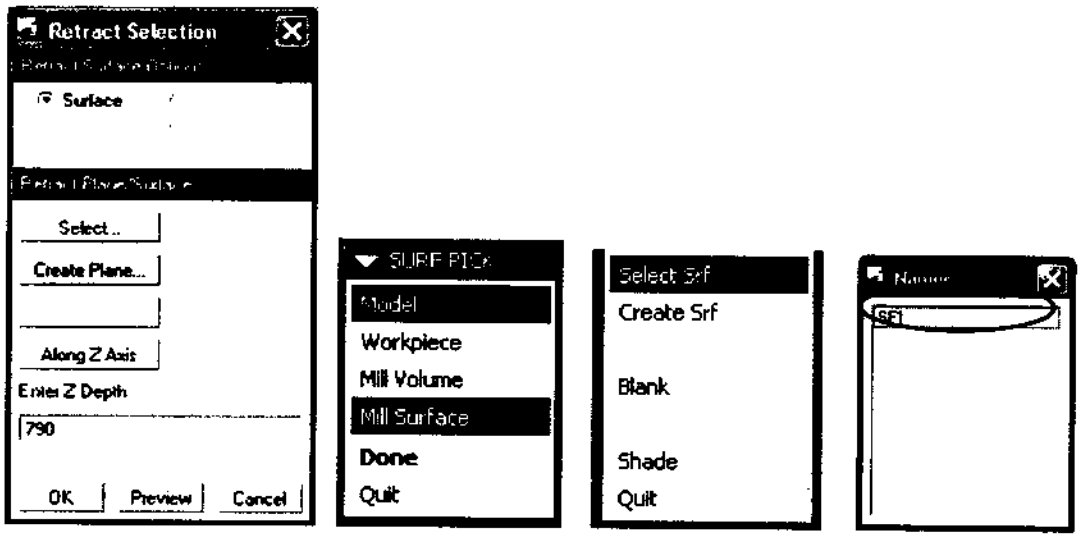
>>**Preview / Apply** >> จากนั้นเมื่อได้แล้ว ต้องการออกไปที่ File / **Done**

รูปที่ 4.28 Tool setup



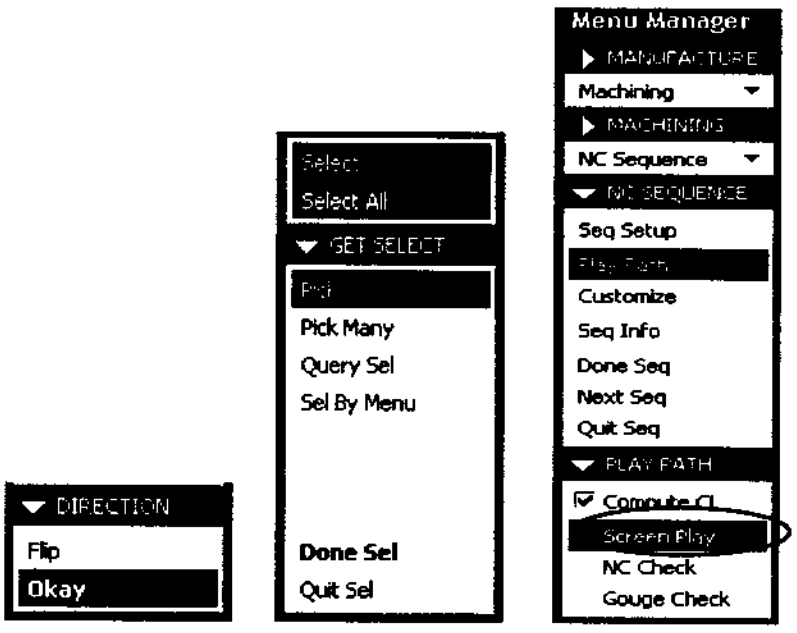
รูปที่ 4.29 (a) Set (b) Param Tree

ในรูป (a) นำ mouse ไป Click ที่ Set จะปรากฏ Param Tree ขึ้นมาให้ Set Parameters โดยจะใส่ค่า cut_feed, step_over, spindle_speed, clear dist / click ***Done*** หรือไปที่ File / exit



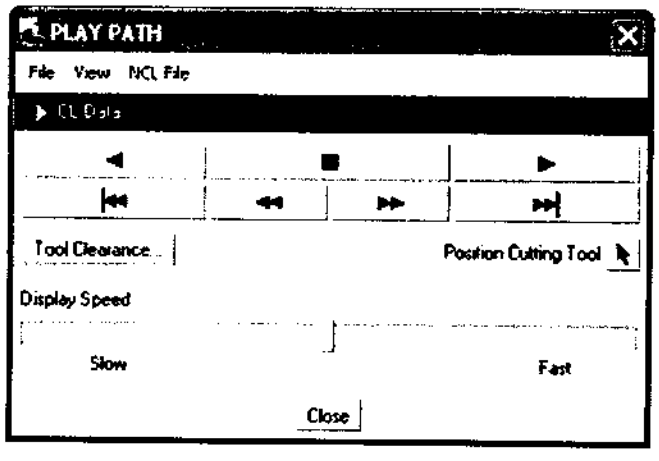
รูปที่ 4.30 Set ค่าระยะของ Tool และ การเลือกผิว

จะปรากฏหน้าจอ Restraction Selection เพื่อกำหนดระยะที่ tool เริ่มเคลื่อนที่ click ***Along Z axis*** /ใส่ ***790*** / click ***OK / Mill Surface / Done /Select Srf /Okay /*** click ***SF1/ Select All***



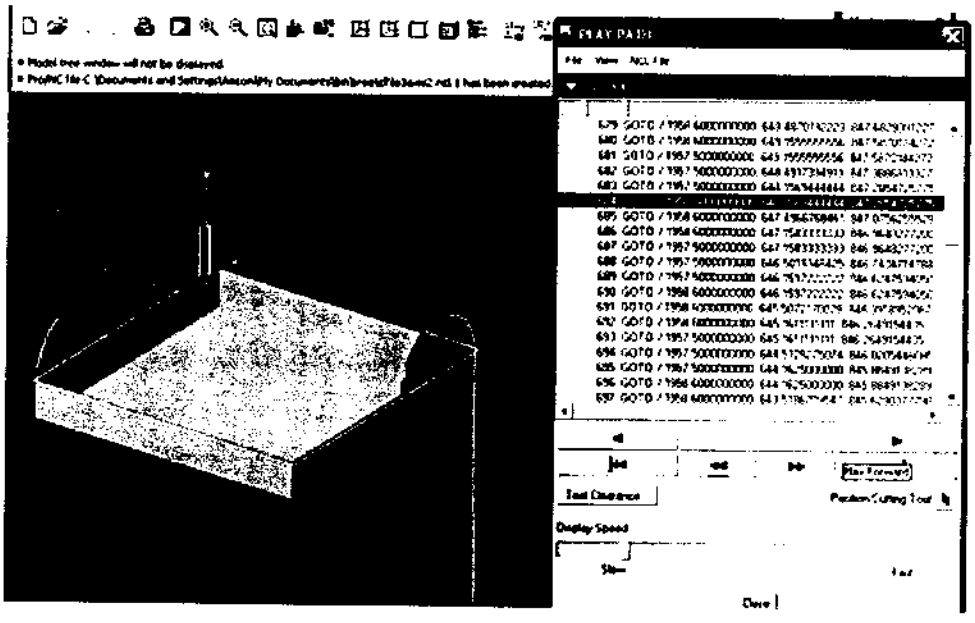
รูปที่ 4.31 การคำนวณ ขนาดของชิ้นงาน

จาก Click Play Path / ☑ Compute CL / Screen Play จะมีการคำนวณ ขนาดของชิ้นงาน ทางเดินของชิ้นงาน



รูปที่ 4.32 CL Data Play Path

สามารถจำลองการเคลื่อนที่ได้ไปตามชิ้นงานที่ได้วางไว้ file ที่ออกมาจะเป็น CL Data ซึ่งเป็นพิกัดของจุดต่าง ๆ ที่ได้เคลื่อนที่ไปดังรูป 4.33



รูปที่ 4.33 แสดงการจำลองการเคลื่อนที่ของ Tool

ในส่วนของโปรแกรม Proengineer 2000i^{2}นี้จะมีการ set แกนได้ 3 axis 4 axis 5 axis อยู่ที่ Manufacturing Setup โดยเข้าไปที่ ***Mfg Setup*** (Machine Tool Setup) จะสามารถกำหนดแกนได้ หลังจากนั้นทำตามขั้นตอนที่กล่าวมาแต่จะมีอยู่อีกอย่างหนึ่งที่สำคัญก็คือตอนเข้า **Machining** > NC Sequence > Surface Mill / เลือก 5 axis / Done จะทำให้สามารถแสดงการเคลื่อนที่ของ โปรแกรม Proengineer 2000 i^{2}บน CL Data Play Path ได้แนวการเคลื่อนที่ออกมาเป็น X , Y , Z , i , j , k ออกมา ในส่วนของขั้นตอนต่อไปจะกล่าวถึงการแปลง file เข้าสู่ หุ่นยนต์อุตสาหกรรม

4.4 การเชื่อมต่อข้อมูลระบบพิกัดตำแหน่งจากโปรแกรม CAD/CAM เข้าสู่ระบบควบคุมและสั่งงานหุ่นยนต์อุตสาหกรรม KUKA Robot - KR125 และผลการทดสอบการเชื่อมต่อข้อมูลพิกัดตำแหน่ง

การนำข้อมูลทางเดินของเครื่องมือมาสร้างโปรแกรมควบคุมหุ่นยนต์อุตสาหกรรม

1.หลังจากได้ไฟล์ข้อมูลทางเดินของเครื่องมือเป็น CL Data ที่มีรูปแบบของข้อมูลดังนี้

```
$$*       Pro/CLfile  Version 2000i2 - 2000060
$$-> MFGNO / MA002
PARTNO / WP2
$$-> FEATNO / 159
MACHIN / UNCX01, 1
$$-> CUTCOM_GEOMETRY_TYPE / OUTPUT_ON_CENTER
UNITS / INCHES
LOADTL / 1
$$-> CUTTER / 10.000000
$$-> CSYS / 1.0000000000, 0.0000000000, 0.0000000000, 0.0000000000, $
        0.0000000000, 1.0000000000, 0.0000000000, 0.0000000000, $
        0.0000000000, 0.0000000000, 1.0000000000, 0.0000000000
SPINDL / RPM, 2000.000000,  CLW
RAPID
GOTO / 1957.5000000000, 911.0000000000, 790.0000000000
FEDRAT / 1000.000000,  IPM
GOTO / 2251.3000000006, 911.0000000000, 790.0000000000
GOTO / 2251.3000000003, 910.1142857143, 790.0000000000
GOTO / 1957.5000000000, 910.1142857143, 790.0000000000
GOTO / 1957.5000000000, 909.2285714286, 790.0000000000
GOTO / 1958.6999999911, 909.2285714286, 790.0000000000
GOTO / 1958.6999999911, 908.3428571429, 790.0000000000
GOTO / 1957.5000000000, 908.3428571429, 790.0000000000
GOTO / 1957.5000000000, 907.4571428571, 790.0000000000
GOTO / 1958.6999999911, 907.4571428571, 790.0000000000
GOTO / 1958.6999999911, 906.5714285714, 790.0000000000
GOTO / 1957.5000000000, 906.5714285714, 790.0000000000
```

GOTO / 1957.5000000000, 905.6857142857, 790.0000000000

GOTO / 1958.6999999911, 905.6857142857, 790.0000000000

GOTO / 1958.6999999881, 904.8000000000, 790.0000000000

GOTO / 1957.5000000000, 904.8000000000, 790.0000000000

GOTO / 1957.5000000000, 903.8013888889, 790.0000000000

GOTO / 1958.6999999881, 903.8013888889, 790.0000000000

GOTO / 1958.6999999881, 902.8027777778, 790.0000000000

GOTO / 1957.5000000000, 907.4571428571, 790.0000000000

GOTO / 1958.6999999911, 907.4571428571, 790.0000000000

GOTO / 1958.6999999911, 906.5714285714, 790.0000000000

GOTO / 1957.5000000000, 906.5714285714, 790.0000000000

GOTO / 2256.4000000000, 914.4131496375, 803.7078061493

GOTO / 1957.5000000000, 907.4571428571, 790.0000000000

GOTO / 1958.6999999911, 907.4571428571, 790.0000000000

GOTO / 1958.6999999911, 906.5714285714, 790.0000000000

GOTO / 1957.5000000000, 906.5714285714, 790.0000000000

GOTO / 2256.4000000000, 914.5516620052, 803.1035444225

GOTO / 2256.4000000000, 914.6841531149, 802.4968065193

GOTO / 1957.5000000000, 907.4571428571, 790.0000000000

GOTO / 1958.6999999911, 907.4571428571, 790.0000000000

GOTO / 1958.6999999911, 906.5714285714, 790.0000000000

GOTO / 1957.5000000000, 906.5714285714, 790.0000000000

GOTO / 1957.5000000000, 907.4571428571, 790.0000000000

...

...

....

GOTO / 2256.4000000000, 914.8105531343, 801.8877600935

GOTO / 2256.4000000000, 914.9307972699, 801.2765745299

GOTO / 2256.4000000000, 914.9989600000, 800.9136651779

GOTO / 2256.4000000000, 914.9989600000, 800.9143354760

GOTO / 2257.5000000000, 914.9989600000, 800.9143354760

SPINDL / OFF

$$-> END /

FINI

ทำการแก้ไขและตัดส่วนที่ไม่สำคัญทิ้งในส่วนที่ดี กรอบ

```
$$*        Pro/CLfile  Version 2000i2 - 2000060
$$-> MFGNO / MA002
PARTNO / WP2
$$-> FEATNO / 159
MACHIN / UNCX01, 1
$$-> CUTCOM_GEOMETRY_TYPE / OUTPUT_ON_CENTER
UNITS / INCHES
LOADTL / 1
$$-> CUTTER / 10.000000
$$-> CSYS / 1.0000000000, 0.0000000000, 0.0000000000, 0.0000000000,  $
        0.0000000000, 1.0000000000, 0.0000000000, 0.0000000000,  $
        0.0000000000, 0.0000000000, 1.0000000000, 0.0000000000
SPINDL / RPM, 2000.000000,  CLW
```

```
RAPID
```

```
GOTO / 1957.5000000000, 911.0000000000, 790.0000000000
```

```
FEDRAT / 1000.000000,  IPM
```

```
GOTO / 2251.3000000006, 911.0000000000, 790.0000000000
GOTO / 2251.3000000003, 910.1142857143, 790.0000000000
GOTO / 1957.5000000000, 910.1142857143, 790.0000000000
...
...
...
GOTO / 2257.5000000000, 914.9989600000, 800.9143354760
```

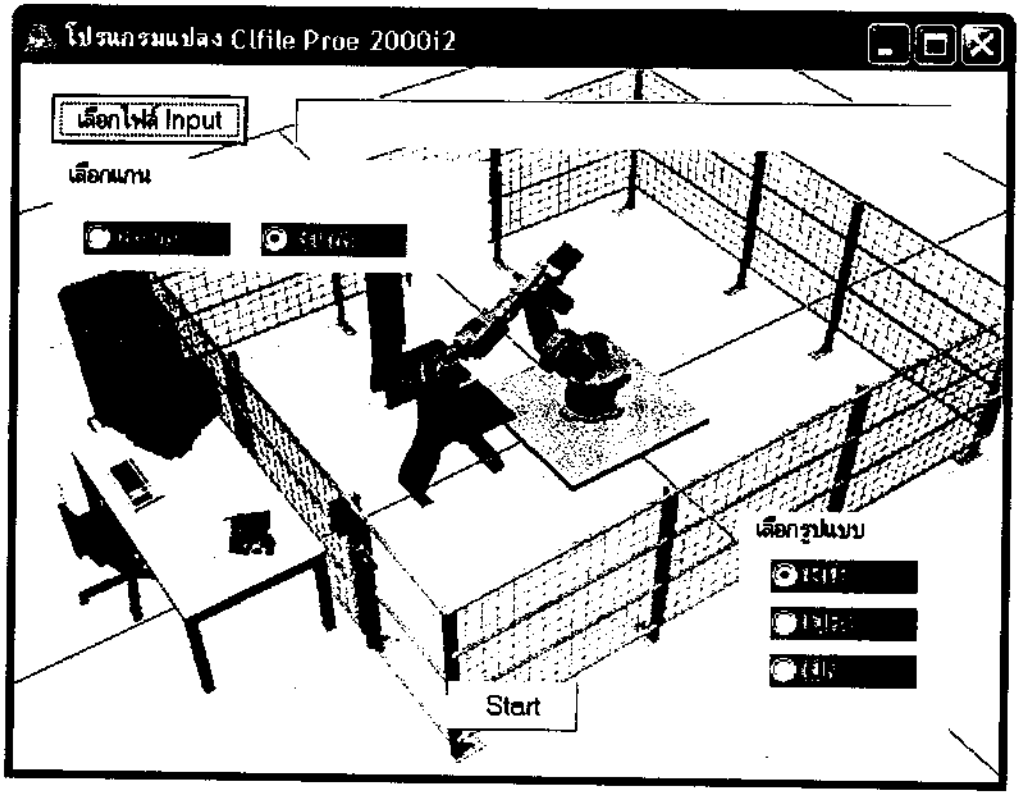
```
SPINDL / OFF
$$-> END /
FINI
```

โดยการกำหนดให้ Visual Basic ที่ได้ Download ตัวที่ Link กับ file text ได้มาเป็น Source Code และทำการเขียนโปรแกรมต่อไป

- ทำการเขียนดัดแปลงและแก้ไข Source Code ด้วยโปรแกรม Visual Basic

การแปลงข้อมูลโดยใช้โปรแกรม Visual Basic 6.0 โดยมีวิธีดังนี้คือ

- เลือกไฟล์ Input ที่มีfile *.ncl.1 โดยการ Brow หาไปยังที่อยู่ของ file
- ทำการเลือกว่าต้องการ 6 แกน หรือ 3 แกน และเลือกรูปแบบ PTP CIRC LIN
- Click ที่ปุ่ม Start



รูปที่ 4.34 แสดงหน้าจอของโปรแกรมแปลงข้อมูล

รายละเอียด Code ดูได้จากภาคผนวก ข.

3. นำข้อมูลทางเดินของเครื่องมือที่แปลงได้ มาเพิ่มส่วนต้น – ส่วนท้าย เพื่อที่จะสามารถนำไปใช้ควบคุมอุตสาหกรรม โดยหลังจากที่เพิ่มส่วนต้น – ส่วนท้าย จะมีรายละเอียดดังนี้

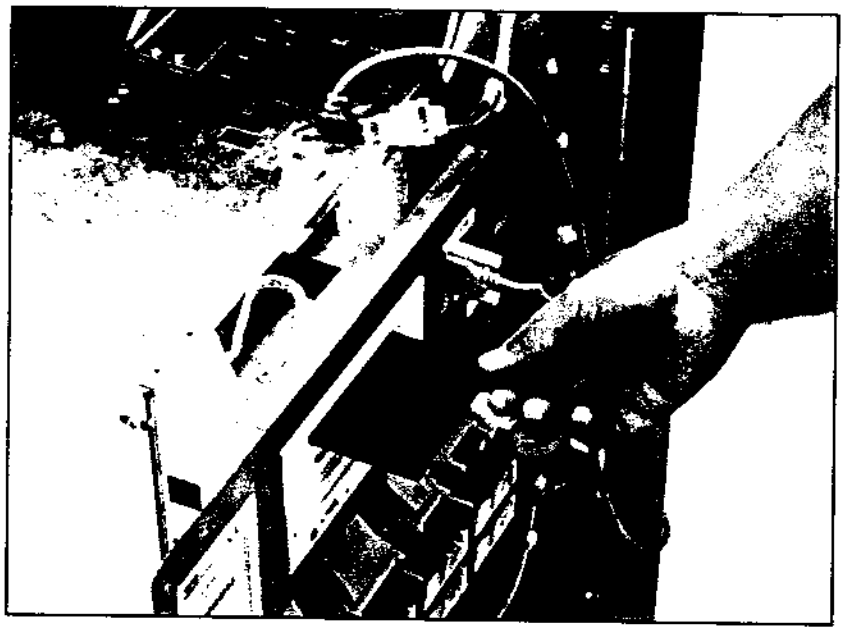
```
DEF AN001()
DECL AXIS HOME
BAS (#INITMOV,0)
HOME ={AXIS:A1 0,A2 -90,A3 90,A4 0,A5 0,A6 0}
PTP HOME

PTP {X 1957.5000000000,Y  911.0000000000,Z  790.0000000000}
PTP {X 2251.3000000006,Y  911.0000000000,Z  790.0000000000}
PTP {X 2251.3000000003,Y  910.1142857143,Z  790.0000000000}
PTP {X 1957.5000000000,Y  910.1142857143,Z  790.0000000000}
PTP {X 1957.5000000000,Y  909.2285714286,Z  790.0000000000}
...
...
...
PTP {X 2256.4000000000,Y  914.9989600000,Z  800.9136651779}
PTP {X 2256.4000000000,Y  914.9989600000,Z  800.9143354760}
PTP {X 2257.5000000000,Y  914.9989600000,Z  800.9143354760}
PTP HOME
END
```

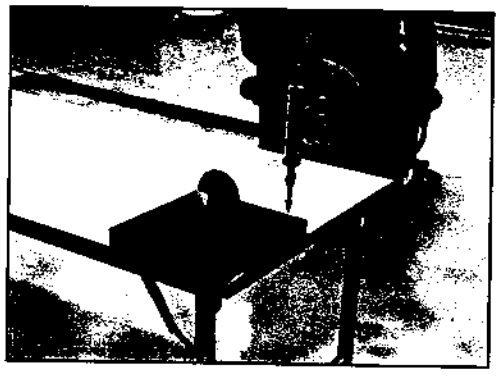
หลังจากเพิ่มส่วนต้น – ส่วนท้ายแล้ว ทำการ copy เสร็จแล้วนำไปวางในโปรแกรม Notepad และบันทึกไฟล์ที่เขียนให้มีรูปแบบดังนี้

- File name = *.SRC
- Save as type = All Files [*.*]

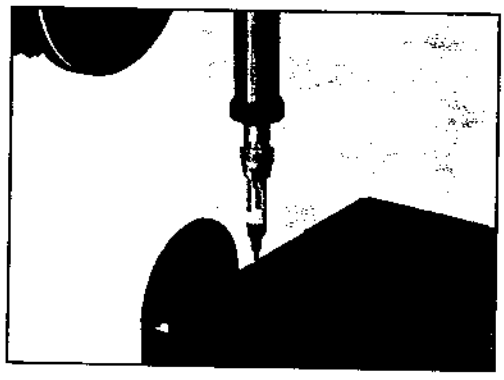
4. นำโปรแกรมที่ได้ป้อนเข้าไปใน Robot จะได้ผลการทดสอบดังรูปที่ 4.35 - 4.37



รูปที่ 4.35 แสดงการใส่แผ่นบันทึกโปรแกรม เข้าสู่ชุดควบคุม

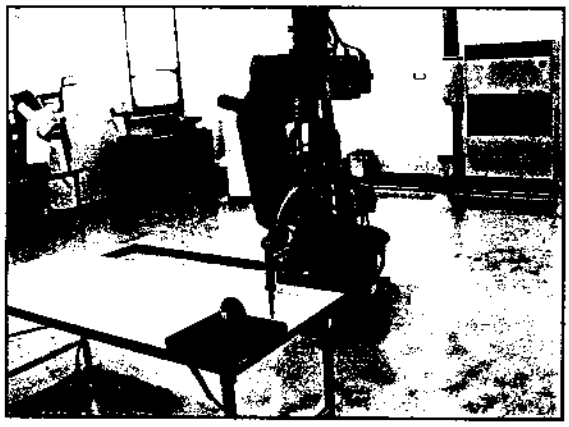


(a)

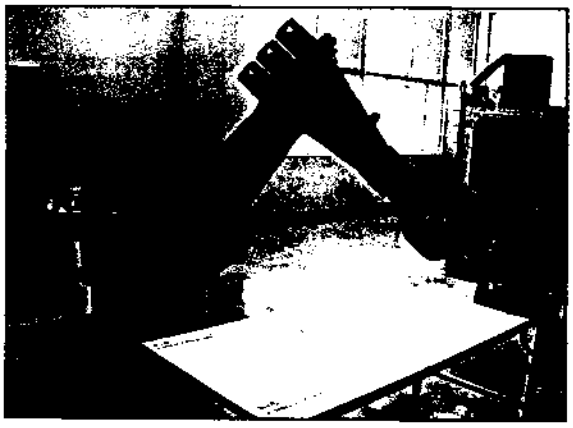


(b)

รูปที่ 4.36 (a), (b) แสดง tool ของ Robot ขณะกำลังเคลื่อนที่บนชิ้นงาน

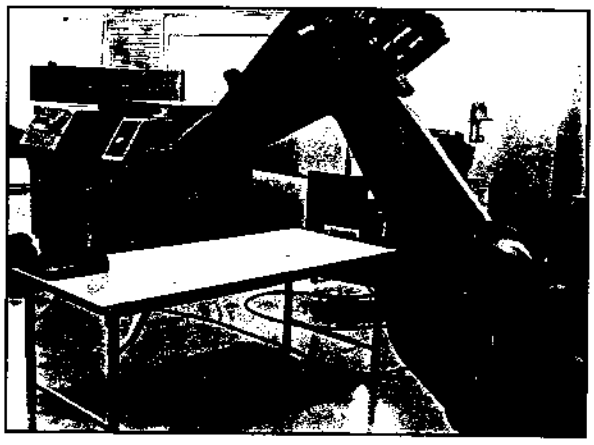


(a)

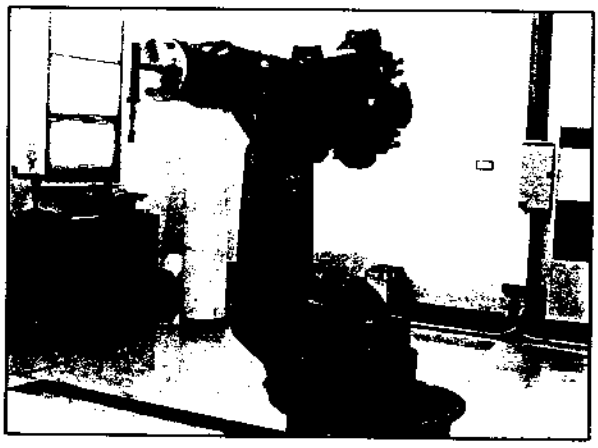


(b)

รูปที่ 4.37 (a), (b) แสดง Robot ขณะกำลังเคลื่อนที่บนชิ้นงาน



(a)



(b)

รูปที่ 4.38 (a), (b) แสดง Robot ขณะกำลังเคลื่อนกลับสู่ Home

ตัวอย่าง Code CL Data 5 - Axis

```
$$*        Pro/CLfile  Version 2000i2 - 2000060
$$-> MFGNO / MFG0003
PARTNO / WP001
$$-> FEATNO / 45
MACHIN / UNCX01, 1
$$-> CUTCOM_GEOMETRY_TYPE / OUTPUT_ON_CENTER
UNITS / INCHES
LOADTL / 1
$$-> CUTTER / 10.000000
$$-> CSYS / 1.0000000000, 0.0000000000, 0.0000000000, 0.0000000000,  $
        0.0000000000, 1.0000000000, 0.0000000000, 0.0000000000,  $
        0.0000000000, 0.0000000000, 1.0000000000, 0.0000000000
MULTAX / ON
SPINDL / RPM, 2000.000000,  CLW
RAPID
GOTO / 2257.5000000000, 791.2072545053, 790.0000000000,  $
0.0000000000, -0.9965457582, 0.0830454799
RAPID
GOTO / 2257.5000000000, 791.2072545053, 794.9827287912,  $
0.0000000000, -0.9965457582, 0.0830454799
FEDRAT / 1000.000000,  IPM
GOTO / 2257.5000000000, 791.2679533153, 795.6641503213,  $
0.0000000000, -0.9955341114, 0.0944025054
...
GOTO / 2256.4000000000, 910.9903843798, 791.0741424314,  $
0.0000000000, 0.9998397397, 0.0179023739
SPINDL / OFF
$$-> END /
FINI
```

จากตัวอย่าง Code CL Data 5 – Axis การเขียนไม่ได้ต่างไปจากเดิมมากนักเพียงแค่การเลือกในการเขียนในส่วนของ Part Manufacturing นั้นจะมีให้เลือกว่าเป็น 3 แกน 4 แกน 5 แกน โดยจะทำบนผิว Surface ในขั้นต่อไปที่มีการเปลี่ยนแปลงคือ Surface Mill นั้นต้องเลือกเป็น 5 แกน จะทำให้เป็นการ set เครื่องมือเพื่อทำงานต่อไปจากนั้นก็จะได้แกนทั้งหมด 5 แกนการทำงานก็จะทำต่อไปได้ดังที่ได้อธิบายไว้ข้างต้น จนจบกระบวนการได้ CL Data ออกมาดังตัวอย่างเราลอง Simulate Part ดูว่าการเคลื่อนที่ถูกต้องหรือเปล่าคำสั่งหรือรูปแบบจะกล่าวไว้แล้วข้างต้นว่าทำอย่างไร

การแปลง File เพื่อนำเข้าสู่ หุ่นยนต์อุตสาหกรรมทำได้โดยใช้โปรแกรม Microsoft Visual Basic 6 เข้าช่วยทำให้แปลงและตัดส่วนที่ไม่จำเป็นทิ้งในส่วนของ การเคลื่อนที่ 6 แกนยังไม่สามารถทำได้เพราะติดปัญหาทางด้านการคิดค้น สมการเพื่อที่จะนำมาแปลงในรูปของ i j k เพราะเป็นสมการในรูปของ Vector โดยต้องหาความสำพันธ์เพื่อ Link เข้าหากันให้ได้ระหว่าง CNC กับ Robot ในงานอุตสาหกรรม

เพื่อเป็นแนวทางต่อไป

$$[X \quad Y \quad Z] \; : \; [i \quad j \quad k]$$

เพราะการเคลื่อนที่ของหุ่นยุนต์ $i = A$, $j = B$, $k = C$ โดย A B C เป็นค่าของมุม นั่นคือ องศา หมายเหตุ ถ้าไม่เข้าใจเนื้อหาหรือว่าไม่เข้าใจอะไรสามารถติดต่อผู้เขียนได้โดยดูจากประวัติผู้เขียนได้จากท้ายเล่ม