การสร้างต้นแบบส่วนควบคุมเครื่องกัดแนวดิ่งซีเอ็นซี

COMPUTERIZED NUMERICAL CONTROL FOR VERTICAL MILLING MACHINE

T030941

รวิ อุตตมธนินทร์

RAVI UTTAMATANIN

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมไฟฟ้า
บัณฑิตวิทยาลัย
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
พ.ศ.2541

ISBN 974-622-194-9

เลขหมู่.....
เลขทะเบียน 30941
วัน, เดือน, ปี 1 ก.ย. 2541

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

COMPUTERIZED NUMERICAL CONTROL FOR VERTICAL MILLING MACHINE

RAVI UTTAMATANIN

A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENT FOR THE DEGREE OF
MASTER OF ENGINEERING IN ELECTRICAL ENGINEERING
SCHOOL OF GRADUATE STUDIES
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG
1998

ISBN 974-622-194-9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

COPYRIGHT 1998

SCHOOL OF GRADUATE STUDIES

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อวิทยานิพนธ์	การสร้างต้นแบบส่วนควบคุมเครื่องกัดแนวดิ่งซีเอ็นซี
นักศึกษา	นายรวิ อุตตมธนินทร์
อาจารย์ผู้ควบคุมวิทยานิพนธ์	ศ.ดร.ไพรัช ธัชยพงศ์
หลักสูตร	วิศวกรรมศาสตร์มหาบัณฑิต
สาขาวิชา	วิศวกรรมไฟฟ้า
พ.ศ.	2541

บทคัดย่อ

การวิจัยและพัฒนาต้นแบบส่วนควบคุมเครื่องกัดแนวดิ่งซีเอ็นซี (Computerizes Numerical Control for Vertical Milling Machine) ที่ได้กล่าวถึงในวิทยานิพนธ์ฉบับนี้จะอยู่ในส่วนของการพัฒนาฮาร์ดแวร์ ซึ่งเป็นส่วนเชื่อมต่อระหว่างคอมพิวเตอร์กับเครื่องจักรกล และซอฟท์แวร์ซึ่งเป็นโปรแกรมคอมพิวเตอร์สำหรับควบคุมการเคลื่อนที่ของมอเตอร์ 3 แกน เพื่อนำไปใช้ในการควบคุมเครื่องจักรกลสำหรับผลิตชิ้นงานตามที่ออกแบบไว้ โปรแกรมจะถูกพัฒนาให้เหมาะกับการใช้งานที่สะดวก และได้เขียนโปรแกรมในส่วนการอ่านค่าคำสั่ง G-Code และ M-Code บางส่วน และเพิ่มความสามารถในการอ่านข้อมูลภาพจากโปรแกรมวาดแบบเพื่อให้ใช้งานได้สะดวกขึ้น ดังนั้นประโยชน์ของการวิจัยและพัฒนาจะได้ต้นแบบของเครื่องควบคุมที่มีราคาไม่แพง ทำให้ต้นทุนในการผลิตต่ำ และยังสามารถใช้งานได้อย่างมีประสิทธิภาพ อีกทั้งยังสามารถนำไปประยุกต์ใช้งานในเชิงอุตสาหกรรมได้ด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Thesis Title	Computerized Numerical Control for Vertical Milling Machine
Student	Mr. Ravi Uttamatanin
Thesis Advisor	Prof.Dr.Pairash Thajchayapong
Degree	Master of Engineering in Electrical Engineering
Year	1998

ABSTRACT

This thesis aim is on developing the control prototype of Computerizes Numerical Control (CNC) for vertical milling machine. The prototype is composed of a hardware and software part in controlling 3-axis motor. The hardware part is an interface card while the core of a software part is a G-code and M-code translator. Moreover, for a user-friendly concept, the graphic to file converter is embedded as well. Those a flexibility and low cost of this prototype cause a production cost low and effectively applicable to the industry of Thailand.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงได้อย่างดี ด้วยคำแนะนำและคำปรึกษาเกี่ยวกับแนวทางการทำวิทยานิพนธ์ในทุกๆด้านจาก ศ.ดร.ไพรัช ธัชยพงศ์ ซึ่งเป็นอาจารย์ผู้ควบคุมวิทยานิพนธ์ ผู้วิจัยรู้สึกซาบซึ้งในความอนุเคราะห์จากท่าน และขอกราบขอบพระคุณเป็นอย่างสูง

ขอขอบพระคุณบิดา มารดา และน้องสาวของผู้วิจัยที่ให้กำลังใจอย่างใกล้ชิด และอดทนในการรอคอยการสำเร็จการศึกษาเป็นเวลานานตลอดการศึกษาของผู้วิจัย

ขอขอบคุณเพื่อนๆ พี่ๆ น้องๆ ทุกๆคนที่ช่วยเหลือในการให้กำลังใจ การจัดพิมพ์ การแก้ไข และการตรวจสอบรายงานจนสำเร็จลุล่วง

คุณค่าและประโยชน์อันพึงมีจากวิทยานิพนธ์ฉบับนี้ผู้วิจัยขอมอบแด่ผู้มีพระคุณทุกท่าน

รวิ อุดตมธนินทร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อภาษาไทย I
บทคัดย่อภาษาอังกฤษ II
กิตติกรรมประกาศ III
สารบัญ IV
สารบัญตาราง VII
สารบัญภาพ XI

บทที่ 1. บทนำ 1
1.1 ที่มาและวัตถุประสงค์.....1
1.2 การออกแบบฮาร์แวร์ และ ซอฟท์แวร์ของระบบ.....2
1.2.1 ส่วนของฮาร์ดแวร์ 2
1.2.2 ส่วนของซอฟท์แวร์ 2
บทที่ 2 ทฤษฎีเครื่องซีเอ็นซี 3
2.1 บทนำ 3
2.2 ส่วนประกอบของเครื่องซีเอ็นซีโดยทั่วไป 4
2.2.1 ส่วนเครื่องจักรที่ใช้ผลิตชิ้นงาน 5
2.2.2 ส่วนควบคุม 9
2.2.3 ส่วนโปรแกรมสั่งงานเครื่องจักร 10
2.2.4 ชุดคำสั่งควบคุมเครื่องจักรที่ใช้ในโครงงาน 12
2.3 การทำงานของเครื่องซีเอ็นซี 15
2.3.1 ระบบควบคุมการเคลื่อนที่ของเครื่องมือ.....15
2.3.2 การวัดระยะการเคลื่อนที่ตำแหน่งเครื่องมือ 18
2.3.3 รูปแบบการทำงานของเครื่องมือ 19
2.3.4 ขั้นตอนการสร้างชิ้นงาน 20
บทที่ 3 การออกแบบและสร้างส่วนควบคุม.....21
3.1 บทนำ 21
3.2 ส่วนของระบบการควบคุมมอเตอร์ของเครื่องต้นแบบ 21
3.2.1 การ์ดควบคุมเซอร์โวมอเตอร์ 21

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

หน้า

3.3 ส่วนของการติดต่อ PLC 25
3.3.1 การเชื่อมต่อวงจรด้านเอาต์พุตของการ์ดกับอุปกรณ์ PLC 25
3.3.2 การเชื่อมต่อวงจรควบคุมกับ PLC ทางอินพุตของ 8255 27
3.4 ส่วนติดต่อกับคอมพิวเตอร์ 28
3.5 สรุปในส่วนของฮาร์ดแวร์ 29
บทที่ 4 การออกแบบและสร้างในส่วนของโปรแกรม 30
4.1 ภาพรวมของโปรแกรมที่ออกแบบ 30
4.1.1 โปรแกรมส่วนควบคุมมอเตอร์ 30
4.2.1 โปรแกรมส่วนแสดงภาพ 3 มิติของชิ้นงาน 34
4.1.3 โปรแกรมส่วนติดต่อกับผู้ใช้งาน 42
4.2 โปรแกรมส่วนของการรับข้อมูลชิ้นงานและเครื่องมือ.....44
4.2.1 การวาดแบบของชิ้นงาน 45
4.2.2 การรับข้อมูลไฟล์จากโปรแกรมจำพวก CAD 45
4.2.2.1 เส้นตรง 46
4.2.2.2 เส้นโค้ง 47
4.2.2.3 วงกลม 49
4.2.2.4 เส้นต่อเนื่อง 50
4.3 การแปลงข้อมูลภาพชิ้นงานเป็นโปรแกรมสั่งงานเครื่องซีเอ็นซี 55
4.4 โปรแกรมส่วนของการจำลองการทำงาน 60
4.5 สรุป 60
บทที่ 5 ตัวอย่างการใช้ซอฟท์แวร์ 61
5.1 คำสั่งที่ใช้โดยทั่วไปในซอฟท์แวร์ 61
5.1.1 คำสั่งไฟล์.....61
5.1.2 คำสั่งรายละเอียดชิ้นงาน และ เครื่องมือ 62
5.1.3 คำสั่งทำงานส่วนโปรแกรมควบคุมเครื่องจักร.....62
5.1.4 คำสั่งจำลองการทำงาน 63
5.1.5 คำสั่งตั้งค่าเครื่องจักร 63
5.1.6 คำสั่ง Jog Mode 64

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

หน้า

5.1.7 คำสั่งทำงานเครื่องจักร 64
5.2 วิธีใช้งานโปรแกรม 66
บทที่ 6 การทดลอง สรุป และ วิจารณ์ผลการทดลอง.....74
6.1 การทดลอง 74
6.1.1 การทดลองส่วนการจำลองการทำงาน 74
6.1.2 ทดลองกัดงานโดยการสั่งผ่านการ์ดควบคุมการทำงาน 76
6.2 สรุปผลการทดลอง 82
6.2.1 ส่วนของฮาร์ดแวร์ 82
6.2.2 ส่วนของซอฟท์แวร์ 82
6.3 ข้อเสนอแนะ 83
บรรณานุกรม.....84
ภาคผนวก 85
ประวัติผู้เขียน 198

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่ **หน้า**

2.1 แสดงชุดคำสั่งที่มีใช้ในโปรแกรม 12
2.2 แสดงชุดคำสั่ง M (M Function) 14
3.1 แสดงการใช้งานพอร์ตต่างๆ ที่เชื่อมต่อเข้ากับคอมพิวเตอร์ 28
6.1 การกำหนดชั้นความลึกงานกัด 75
6.2 แสดงชั้นความลึกของชิ้นงานทดลอง 78

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ

ภาพที่		หน้า
2.1	แสดงเครื่องกัดแนวดิ่งซีเอ็นซี	3
2.2	แสดงเครื่องซีเอ็นซีในรูปแบบต่างๆ	5
2.3	แสดงโต๊ะงาน บอลล์สกรู และ การขับป้อนโต๊ะงาน	6
2.4	แสดงมอเตอร์ไฟฟ้าแบบต่างๆ	7
2.5	แสดงการวัดตำแหน่งโดยตรง	8
2.6	แสดงการวัดตำแหน่งทางอ้อม	8
2.7	แสดงเพลางานของเครื่องกัด	9
2.8	แสดงระบบควบคุมเครื่องจักรกลซีเอ็นซี	9
2.9	แสดงการใช้คอมพิวเตอร์ส่วนบุคคลควบคุมเครื่องจักรกลซีเอ็นซี	9
2.10	แสดงผลของคำสั่งในโปรแกรมเอ็นซี	10
2.11	แสดงโปรแกรมบล็อกที่ประกอบด้วยคำ 7 คำ	11
2.12	แสดงการควบคุมแบบจุดต่อจุด	16
2.13	แสดงการควบคุมแบบตัดตรง	16
2.14	แสดงการควบคุมแบบตามเส้นขอบรูป	17
2.15	แสดงการใช้สเกลวัดตำแหน่งแบบสัมบูรณ์	18
2.16	แสดงสเกลการวัดตำแหน่งแบบต่อเนื่อง	19
2.17	แสดงลักษณะการสร้างชิ้นงาน	20
3.1	แสดงลักษณะการควบคุมของ HCTL-1100	22
3.2	แสดงระบบควบคุมมอเตอร์ของการ์ดควบคุม	23
3.3	แสดงบล็อกไดอะแกรมของการ์ด	23
3.4	แสดงวงจรย่อยภายในการ์ดควบคุมที่ใช้ HCTL-1100	24
3.5	แสดงบล็อกไดอะแกรมของการติดต่อกับ PLC	25
3.6	แสดงการต่อ 8255 กับ Op-to ทางด้านเอ้าพุท	26
3.7	แสดงการต่อ 8255 กับ Op-to ทางด้านอินพุท	27
4.1	แสดงการกำหนดการเคลื่อนที่เป็นฟังก์ชั่นเส้นตรงเชิงความเร็ว	32
4.2	แสดงการประมาณส่วนของวงกลมด้วยเส้นตรง	34
4.3	แสดงจุดในระบบพิกัดฉาก 3 มิติ	35

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ (ต่อ)

ภาพที่		หน้า
4.4	แสดงการหมุนของจุด รอบระบบแนวแกนพิกัดฉาก	37
4.5	แสดงการฉายเงาภาพ	37
4.6	แสดงภาพที่ได้จากการทำโปรเจคชั่น	38
4.7	แสดงวัตถุในระบบออปเจ็กโคออร์ดิเนท	39
4.8	แสดงการเปลี่ยนจุดอ้างอิงภาพ	39
4.9	แสดงการเปลี่ยนเวคเตอร์ตั้งฉากของระนาบภาพ	40
4.10	แสดงการเปลี่ยนระยะภาพ	40
4.11	แสดงการเปลี่ยนวิวอัปเวคเตอร์	41
4.12	แสดงการเปลี่ยนทิศทางโปรเจคชั่น	41
4.13	แสดงโปรแกรมส่วนสร้างโค้ดควบคุมเครื่องจักร	44
4.14	แสดงรูปแบบการเก็บส่วนข้อมูลที่เป็นองค์ประกอบของภาพในไฟล์ dxf	45
4.15	แสดงแบบการเก็บข้อมูลเส้นตรงของไฟล์ dxf	46
4.16	แสดงตัวอย่างข้อมูลเส้นตรงในรูปแบบ dxf	47
4.17	แสดงแบบการเก็บข้อมูลเส้นโค้งของไฟล์ dxf	48
4.18	แสดงตัวอย่างของส่วนโค้งในรูปแบบ dxf	48
4.19	แสดงรูปแบบการเก็บข้อมูลวงกลมของไฟล์ dxf	49
4.20	แสดงตัวอย่างของวงกลมในรูปแบบของ dxf	49
4.21	แสดงข้อมูลภาพเส้นต่อเนื่องทั้งหมด และข้อมูลในส่วนจุดเชื่อมต่อ ของแต่ละเส้น	50
4.22	แสดงการคำนวณหาค่าความนูน (BULGE) ของส่วนโค้ง	51
4.23	แสดงแบบภาพเส้นต่อเนื่อง	53
4.24	แสดงตัวอย่างข้อมูลภาพต่อเนื่อง	54
4.25	แสดงสถานะของเส้นแบบต่างๆ ที่อยู่บนเส้นรอบรูปวงปิด	56
4.26	แสดงวิธีการหาสถานะของเส้นที่อยู่บนรูปวงปิด	56
4.27	แสดงเส้นทางเดินของเครื่องมือภายในเส้นรอบรูปวงปิด	57
4.28	แสดงเส้นทางเดินของเครื่องมือภายนอกเส้นรอบรูปวงปิด	58
4.29	แสดงค่าออฟเซ็ตในแนวแกน X และ Y ที่เกิดจากเส้นทางเดินและเครื่องมือ	59
4.30	แสดงแนวเส้นทางเดินของแนวกัดภายในเส้นรอบรูปวงปิด	60

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ (ต่อ)

ภาพที่		หน้า
5.1	แสดงเมนูการทำงานไฟล์ข้อมูล	61
5.2	แสดงเมนูรายละเอียดชิ้นงาน และ เครื่องมือ	62
5.3	แสดงเมนูทำงานส่วนโปรแกรมคุมเครื่องจักร	62
5.4	แสดงคำสั่งจำลองการทำงาน	63
5.5	แสดงคำสั่งสำหรับตั้งค่า	63
5.6	แสดงคำสั่ง Jog Mode	64
5.7	แสดงคำสั่งทำงานเครื่องจักร	64
5.8	แสดงเมนูการทำงานของโปรแกรม	65
5.9	แสดงภาพชิ้นงานที่สร้างขึ้นเป็น Dxf ไฟล์ของ Acad R.12	66
5.10	แสดงการใช้คำสั่งไฟล์ และเลือกอ่านข้อมูลภาพด้านบน Dxf file	66
5.11	แสดงการเลือกไฟล์ Dxf ที่ต้องการ	67
5.12	แสดงการโหลดไฟล์ Dxf เข้าสู่โปรแกรมแล้ว	67
5.13	การใช้คำสั่งชิ้นงานเครื่องมือ และเลือกชิ้นงานใหม่	68
5.14	แสดงการกำหนดขนาดของชิ้นงานที่ใช้ในการกัดในโปรแกรม	68
5.15	แสดงการใช้คำสั่งชิ้นงาน เครื่องมือ และเลือกเครื่องมือ	69
5.16	แสดงการใส่ค่าเครื่องมือที่ใช้งานลงไป	69
5.17	แสดงการใช้คำสั่งโปรแกรมเครื่องจักร และเลือกกำหนดชั้นความลึกชิ้นงาน	70
5.18	แสดงการใส่ค่าชั้นความลึกชิ้นงาน	70
5.19	การใช้คำสั่งโปรแกรมควบคุมเครื่องจักร และเลือกสร้างโปรแกรมควบคุมเครื่อง	71
5.20	แสดงหน้าจอหลังจากการคำนวณหาเส้นทางเดินการกัดแล้ว	71
5.21	แสดงการเลือกเมนูจำลองการทำงาน แล้วแสดงการจำลองการทำงาน	72
5.22	แสดงชิ้นงานจำลองหลังการกัดแล้ว	72
5.23	แสดงการจับยึดวัตถุที่จะกัด และ แสดงการกัด	73
6.1	ภาพงานที่ใช้ทดสอบ	74
6.2	แสดงชิ้นงานจำลองหลังการกัดจากคำสั่งจำลองการทำงาน	75
6.3	แสดงต้นแบบส่วนควบคุมมอเตอร์ที่สร้างขึ้นในโครงงาน	76
6.4	แสดงเครื่องกัดแนวดิ่งซีเอ็นซีที่ใช้ในการทดลอง	76
6.5	แสดงผลจากการสั่งทำงานเครื่องจักรกัดเป็นรูปวงกลม	77

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ (ต่อ)

ภาพที่ **หน้า**

6.6 แสดงรูปงานจากโปรแกรมวาดแบบ 78
6.7 แสดงชิ้นงานหลังจากการกัดโดยโค้ดจากโปรแกรมในโครงงาน 81
6.8 แสดงชิ้นงานหลังการกัดจากการจำลองการทำงาน 82

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ที่มาและวัตถุประสงค์

เนื่องด้วยในส่วนของภาคอุตสาหกรรมเป็นจำนวนมากมีความต้องการที่จะปรับปรุงการผลิตให้ดีขึ้น ทั้งด้านคุณภาพ และ ปริมาณของชิ้นงาน ซึ่งแต่เดิมการผลิตจะใช้เครื่องจักรที่มีความจำเป็นต้องใช้ช่างฝีมือในการทำงาน ทำให้ความสามารถในการสร้างชิ้นงานขึ้นมามีปัจจัยขึ้นอยู่กับหลายอย่าง และอย่างหนึ่งขึ้นอยู่กับฝีมือของช่างแต่ละบุคคล และประสิทธิภาพในการทำงานที่ไม่เท่ากันของช่างแต่ละคน ความเหนื่อยล้าในการทำงาน ทำให้ไม่สามารถทำชิ้นงานที่เหมือนๆ กันในปริมาณมากๆ ได้ และชิ้นงานที่ทำก็อาจมีมาตราฐานที่ไม่เท่ากัน ดังนี้จึงมีผู้สร้างเครื่องจักรที่ใช้คอมพิวเตอร์เข้ามาควบคุมการทำงาน เรียกว่า เครื่องจักรซีเอ็นซี (Computer Numerical Control) โดยการทำงานนั้นจะสามารถโปรแกรมให้เครื่องผลิตชิ้นงานที่ต้องการเป็นจำนวนมากๆ ได้ และ ได้คุณภาพของงานตามที่ต้องการทุกชิ้น และยังสามารถเปลี่ยนรูปแบบของชิ้นงานที่สร้างได้โดยง่าย เพียงแต่อาศัยการเปลี่ยนโปรแกรมการสั่งงานเท่านั้น ทำให้ไม่ต้องอาศัยช่างฝีมือที่ชำนาญการมากนักในการทำงาน เพียงแต่ต้องการช่างที่ควบคุมการทำงานของเครื่องให้ได้เท่านั้น ทำให้การทำงานง่ายขึ้นมาก แต่ว่าเครื่องจักรซีเอ็นซีนั้นเป็นเครื่องจักรที่มีราคาสูงมาก ดังนั้นในอุตสาหกรรมขนาดเล็ก หรือ ขนาดกลางจึงไม่สามารถนำเครื่องซีเอ็นซีมาใช้งานได้ เพราะจะเป็นการเพิ่มต้นทุนซึ่งสูงเกินไป จะทำให้ไม่คุ้มค่าในการลงทุน แต่ในอุตสาหกรรมหลายรูปแบบก็ต้องการเครื่องซีเอ็นซีไปใช้งาน และในระบบอุตสาหกรรมจะมีเครื่องจักรที่ใช้งานเดิมสำหรับทำงานอยู่แล้ว จะพบว่าถ้าเราสามารถปรับปรุงเครื่องจักรเก่าที่ยังใช้งานได้ มาติดตั้งส่วนควบคุม ที่สามารถควบคุมได้ด้วยเครื่องคอมพิวเตอร์ได้ จะทำให้เครื่องจักรตัวนั้นสามารถทำงานได้ในลักษณะเทียบเท่ากับเครื่องซีเอ็นซีราคาแพงได้ จึงมีความต้องการใช้ส่วนควบคุมในรูปแบบดังกล่าวเป็นจำนวนมาก

ในโครงการนี้ได้ออกแบบส่วนควบคุมทั้งในส่วนของฮาร์ดแวร์ และซอฟท์แวร์ โดยในส่วนของฮาร์ดแวร์นั้นทำเป็นการ์ดอินเตอร์เฟสต่อกับเครื่องคอมพิวเตอร์ส่วนบุคคลที่ใช้อยู่โดยทั่วไป และในส่วนของซอฟท์แวร์ได้ออกแบบให้การ์ดนี้ใช้ควบคุมเครื่องจักรที่เป็นเครื่องกัดแนวดิ่ง และส่วนควบคุมที่ออกแบบในโครงงานนี้ ยังสามารถนำไปใช้ควบคุมเครื่องจักรที่ใช้มอเตอร์ควบคุมการทำงานได้ในหลายๆ รูปแบบ เพียงแต่เปลี่ยนในส่วนของซอฟท์แวร์เท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.2 การออกแบบฮาร์ดแวร์ และ ซอฟท์แวร์ของระบบ

การวิจัยนี้เป็นการสร้างส่วนควบคุมเครื่องจักรสำหรับควบคุมเครื่องกัดแนวดิ่ง (Vertical Milling Machine) หรือ เรียกสั้นๆ ว่า เรื่องซีเอ็นซี โดยส่วนควบคุมที่สร้างขึ้นมานี้จะมีความสามารถในการควบคุมการเคลื่อนที่ของมอเตอร์ในระบบ 3 แกน พร้อมทั้งสามารถควบคุมรีเรย์ รับข้อมูลจากสวิทช์ และควบคุมอัตราเร็วในการหมุนของส่วนเพลางานด้วย

1.2.1 ส่วนของฮาร์ดแวร์

ในโครงงานนี้ได้ออกแบบให้ใช้เครื่องไมโครคอมพิวเตอร์เป็นตัวหลักในการประมวลผลสำหรับควบคุมการทำงาน และได้สร้างเป็นการ์ดอินเตอร์เฟสขึ้นมาโดยเป็นบอร์ดที่มีความสามารถในการควบคุมมอร์เตอร์ได้ 4 ตัว มีส่วนรับสัญญาณ เอ็นโค้ดเดอร์ และมีส่วนติดต่อกับสวิทช์ และรีเรย์ต่างๆ รวมเข้าอยู่ด้วยโดยให้มีระดับสัญญาณในหารควบคุมเท่ากับระบบพีแอลซีมาตราฐาน คือ มีระดับสัญญาณอยู่ที่ 0-24 โวลท์

ในส่วนของการควบคุมมอร์เตอร์ ได้ใช้ซิฟไอซีสำหรับการควบคุมการเคลื่อนที่ HCTL-1100 มาใช้งานเป็นส่วนควบคุมในแต่ละแนวแกน เพื่อลดความยุ่งยากซับซ้อนของวงจร จึงทำให้การออกแบบสร้างปรับแต่ง และการควบคุมการเคลื่อนที่ให้สัมพันธ์กันทั้ง 3 แกนลดความยุ่งยากลงมาก

1.2.2 ส่วนของซอฟท์แวร์

ในส่วนของโปรแกรมในโครงงานนี้แบ่งได้เป็น 2 ส่วนใหญ่ๆ คือ

1 ส่วนซอฟท์แวร์สำหรับติดต่อกับผู้ใช้งาน (User Interface)

2 ส่วนซอฟท์แวร์สำหรับควบคุมการทำงานของเครื่องจักร (Motion and Logic Control)

ส่วนของการติดต่อกับผู้ใช้งานจะเป็นการรับข้อมูลของชิ้นงาน รับข้อมูลของแบบที่จะให้กัด รับข้อมูลความลึกของชิ้นงาน มีการทำการจำลองการทำงานของเครื่องกัด และยังสามารถแสดงภาพ 3 มิติ หลังจากการกัดได้ด้วย

ในส่วนของการควบคุมเครื่องจักรจะเป็นการควบคุมให้เครื่องจักรเคลื่อนที่ได้สัมพันธ์กันให้ได้ชิ้นงานตามที่โปรแกรมไว้

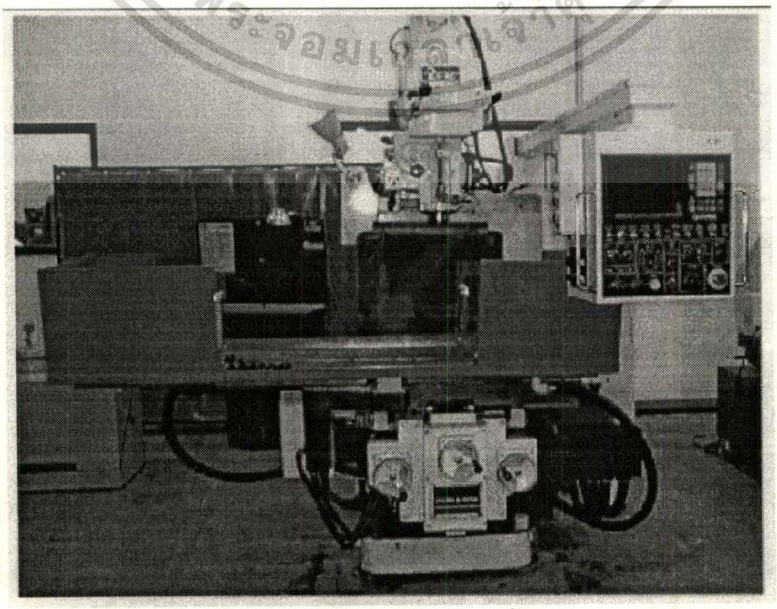
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีเครื่องซีเอ็นซี

2.1 บทนำ

เครื่องจักรซีเอ็นซีเป็นเครื่องจักรที่ใช้คอมพิวเตอร์ในการควบคุมการทำงาน ซึ่งมีหลายรูปแบบแตกต่างกันไปตามลักษณะของการใช้งาน เช่น เครื่องกลึง เครื่องกัด เครื่องคว้าน เครื่องตัดเจาะต่างๆ เป็นต้น โดยเครื่องจักรแบบนี้มีลักษณะเด่นอยู่ว่า จะทำงานตามโปรแกรมสั่งงานที่ตั้งไว้ และยังสามารถทำการแก้ไขปรับเปลี่ยนโปรแกรมการทำงาน เพื่อสร้างชิ้นงานในลักษณะต่างๆ กันออกมาได้โดยง่าย โดยอาศัยเครื่องจักรตัวเดิม ไม่ต้องออกแบบตัวเครื่องจักรใหม่ เพียงแต่ปรับเปลี่ยนโปรแกรมการทำงานเท่านั้น ทำให้มีความสะดวกในการทำงานเพิ่มขึ้นมาก และยังสามารถทำชิ้นงานที่เหมือนๆ กันออกมาได้ตามจำนวนที่ต้องการ ในเวลาที่สามารถกำหนดได้ เป็นการประหยัดทั้งเวลาทำงาน และ แรงงานในการทำงาน และโปรแกรมการทำงานยังเห็นเสมือนการจัดเตรียมการทำงานอย่างเป็นขั้นตอน เป็นการป้องกันความเสียหายที่อาจเกิดขึ้นได้ และ ยังเป็นการช่วยลดเวลาในการผลิต เช่น ลดเวลาในการตรวจสอบขนาดชิ้นงาน ลดเวลาในการปรับเปลี่ยนความเร็วรอบในการทำงาน ทำให้มีความถูกต้องแม่นยำในการทำงานสูงขึ้นมาก

รูปที่ 2.1 แสดงเครื่องกัดแนวดิ่งซีเอ็นซี



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครื่องจักรกลซีเอ็นซีที่นำมาใช้งานโดยทั่วไปนี้จะมีข้อดี ข้อเสีย เปรียบเทียบกับเครื่องจักรกลโดยทั่วๆไปพอสรุปได้ดังนี้

ข้อดีของเครื่องจักรกลซีเอ็นซีมีดังนี้

1 มีความยืดหยุ่นในการทำงานสูง
2 มีความเที่ยงตรงในการทำงานสูง
3 เวลาที่ใช้ในการผลิตจะเร็วกว่า
4 สามารถผลิตชิ้นงานที่มีความซับซ้อนได้ง่ายกว่า
5 การปรับตั้งระบบการทำงานจะง่ายกว่า ใช้เวลาน้อยกว่าการผลิตโดยวิธีอื่นๆ
6 สามารถหลีกเลี่ยงการใช้ช่างควบคุมที่ต้องมีทักษะ และประสพการณ์สูง
7 ช่างควบคุมจะมีเวลาว่างจากการควบคุมเครื่อง ทำให้สามารถจัดเตรียมงานอื่นๆ ต่อไปได้
8 การตรวจสอบคุณภาพจะทำได้ง่ายขึ้น

ข้อเสียของเครื่องจักรซีเอ็นซี คือ

1 ราคาค่อนข้างสูง
2 การบำรุงรักษาซับซ้อนกว่า
3 จำเป็นต้องใช้ช่างเขียนโปรแกรมที่มีทักษะ และถูกฝึกอบรมมาเฉพาะ
4 ชิ้นส่วน หรือ อะไหล่ในการซ่อมบำรุงมีราคาแพง
5 การซ่อมบำรุงต้องอาศัยช่างที่มีประสพการณ์สูง และผ่านการฝึกอบรมมาอย่างดี
6 ราคาของเครื่องมือ(Tool) ต่างๆ ที่ใช้ในขบวนการผลิตมีราคาสูง
7 พื้นที่ที่ใช้ในการติดตั้งเครื่องจักรต้องการการควบคุมที่ดี เช่น ควบคุมอุณหภูมิ ความชื้น ความร้อน

จากข้อดี และข้อเสียที่เห็นจะพบว่าในระบบอุตสาหกรรมขนาดเล็ก จนถึงขนาดกลางนั้น ในส่วนของราคาจะไม่ค่อยคุ้มค่ากับการลงทุน ดังนั้นถ้าหากสามารถดัดแปลงเฉพาะในส่วนของการควบคุมเข้าไปยังเครื่องจักรเก่าที่มีอยู่จะทำให้ราคาของเครื่องซีเอ็นซีที่ได้จะถูกลงมาก และในส่วนของซอฟท์แวร์ที่ออกแบบขึ้นไม่ซับซ้อนทำให้ใช้งานได้โดยง่าย

2.2 ส่วนประกอบของเครื่องซีเอ็นซีโดยทั่วไป

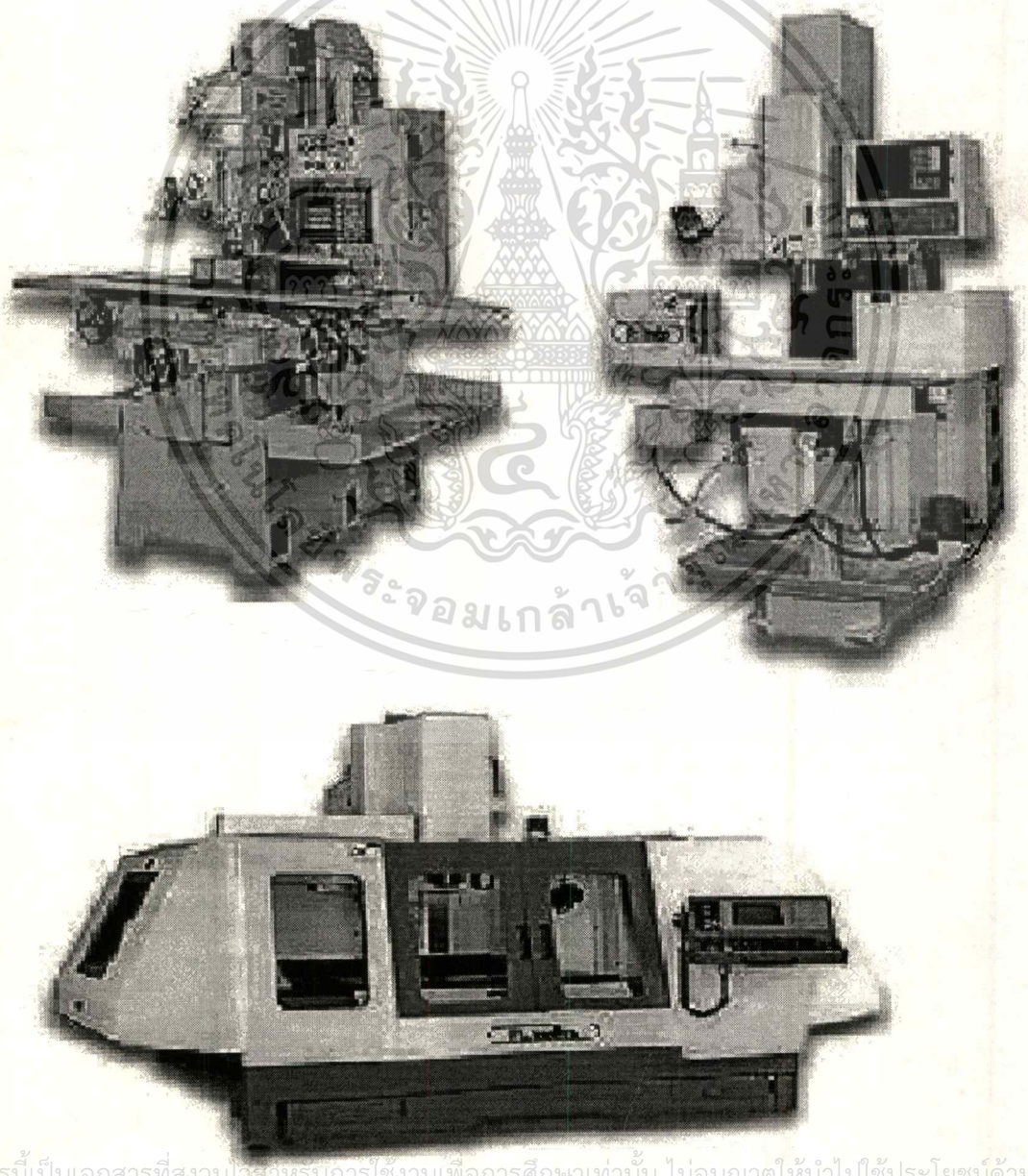
เครื่องซีเอ็นซีนี้จะแบ่งเป็น 2 ส่วนด้วยกัน ดังนี้ ส่วนแรกจะเป็นส่วนของเครื่องจักรที่ใช้สำหรับสร้างชิ้นงานตามต้องการออกมา และ อีกส่วนหนึ่งจะเป็นส่วนของคอมพิวเตอร์ที่ใช้ควบคุมการทำงาน เป็นส่วนที่รับ บันทึก แก้ไข โปรแกรมการทำงาน ที่ทำการควบคุมเครื่องจักรให้ผลิตชิ้นงานต่างๆ ออกมาอีกที โดยเครื่องจักรกลแบบนี้การเคลื่อนที่จะอาศัยมอเตอร์เป็นตัวขับให้เคลื่อนที่ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.1 ส่วนเครื่องจักรที่ใช้ผลิตชิ้นงาน

เครื่องจักรกลซีเอ็นซีจะมีอยู่หลายชนิดมากแต่ในที่นี้จะเป็นชนิดเครื่องกัดแนวดิ่งซีเอ็นซี ซึ่งเครื่องกัดซีเอ็นซีนี้จะเป็นเครื่องจักรกลที่มีขอบเขตการทำงานกว้างขวาง คือ นอกจากจะทำงานเป็นเครื่องกัดโดยทั่วไปแล้ว ยังสามารถทำงานในลักษณะอื่นๆ ได้ เช่น ทำรู เกลียว คว้านรู ได้อีกด้วย และในเครื่องกัดก็ยังมีแบบที่มีแนวแกนควบคุมตั้งแต่ 3 แกน 4 แกน 5 แกน และ มากกว่านั้น ดังรูป

รูปที่ 2.2 แสดงเครื่องซีเอ็นซีในรูปแบบต่างๆ

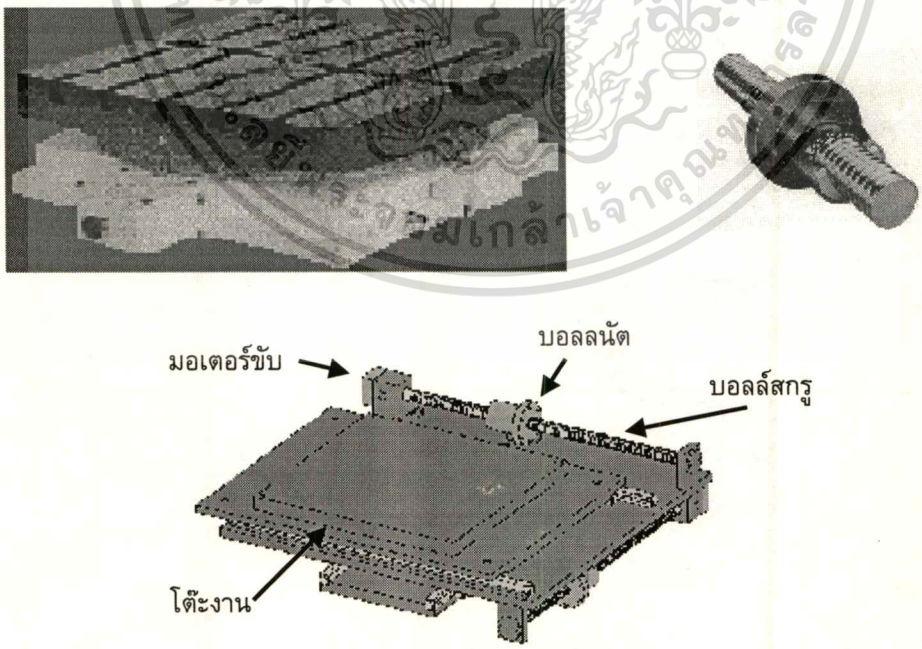


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครื่องกัดนี้จะประกอบไปด้วยส่วนต่างๆ ดังนี้

โต๊ะสำหรับจับยึดชิ้นงาน เป็นโต๊ะที่นำชิ้นงานที่จะทำการกัดไปจับยึดไว้บนนั้น โดยการจับยึดชิ้นงานกัด จุดสำคัญจะอยู่ที่การกำหนดตำแหน่งของชิ้นงานบนโต๊ะงานที่ถูกต้อง และรวดเร็ว ควรเป็นอุปกรณ์ที่สามารถถอด และใส่ชิ้นงานได้รวดเร็ว และสามารถบังคับตำแหน่งในการทำงานของชิ้นงานซ้ำๆได้อย่างถูกต้องเที่ยงตรงเสมอ พร้อมทั้งยังสามารถเลื่อนไปมาตามแนวแกนได้อย่างถูกต้องแม่นยำ โดยอาศัยกำลังขับจากมอเตอร์ไฟฟ้าป้อนขับให้โต๊ะเลื่อนส่งผ่านกำลังขับจากมอเตอร์ผ่านไปยังบอลล์สกรู ซึ่งมีลักษณะเป็นสกรู กับนัต ที่มีลักษณะเป็นเกลียวกลม เมื่อมอเตอร์หมุนขับสกรู นัตก็จะเคลื่อนที่ไปตลอดความยาวของสกรู พาแท่นเลื่อนและโต๊ะงานเคลื่อนที่ไปตามรางเลื่อน ในนัตเป็นเหล็กชุบแข็ง เจียรไนผิวเรียบมัน เพื่อเป็นการลดความฝืด และเพิ่มความถูกต้องในการเคลื่อนที่ และยังประกอบด้วยลูกบอลจำนวนมากหมุนวนอยู่ตลอดภายใน ทำให้ความเสียดทานในการส่งกำลังขับจากสกรูไปยังแท่นเลื่อนมีน้อยมาก และนัตยังออกแบบสำหรับลดระยะคลอนเพื่อให้การเคลื่อนที่ของแท่นเลื่อนมีความเที่ยงตรงสูง ดังนั้นการควบคุมโต๊ะให้เลื่อนไปยังตำแหน่งที่ต้องการก็คือ การควบคุมการหมุนของมอเตอร์ให้หมุนในทิศทาง และความเร็วที่ถูกต้องและแม่นยำ

รูปที่ 2.3 แสดง โต๊ะงาน บอลล์สกรู และ การขับป้อนโต๊ะงาน



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

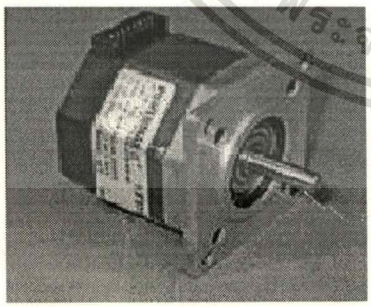
ระบบป้อนขับ โดยทั่วไปใช้มอเตอร์ในการหมุนขับ และ ควบคุมการทำงานด้วยวงจรอิเลคทรอนิคส์ เป็นมอเตอร์ไฟฟ้าที่สามารถควบคุมให้หยุด หรือ ให้หมุนในทิศทางที่ต้องการ และ หมุนในความเร็วใดๆ ที่ต้องการก็ได้ ในระบบซีเอ็นซีสมัยใหม่จะใช้ระบบขับป้อนแบบเซอร์โว (Servo drivers) ทำให้สามารถปรับอัตราป้อน ความเร็วรอบได้อย่างไม่จำกัดตลอดช่วงความเร็วอย่างแม่นยำอย่างรวดเร็ว โดยทั่วๆไปมอเตอร์ที่ใช้ในระบบขับป้อนมี 3 ชนิดด้วยกัน คือ

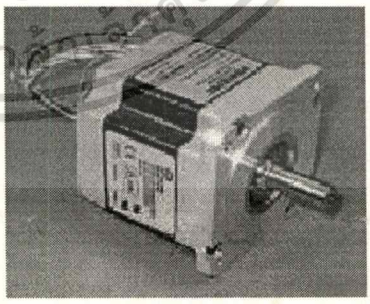
มอเตอร์กระแสตรง (DC Motor) มอเตอร์ชนิดนี้ มีข้อดีคือ เป็นมอเตอร์ที่สามารถปรับอัตราป้อนได้อย่างละเอียด และมีวงจรควบคุมที่ไม่ซับซ้อน แต่มีข้อเสียคือ เมื่อต้องการกำลังขับสูง มอเตอร์จะมีขนาดใหญ่ และเมื่อความเร็วรอบสูงๆ แรงบิดจะลดลง ดังนั้นจะใช้กับเครื่องจักรกลขนาดกลาง และเล็ก

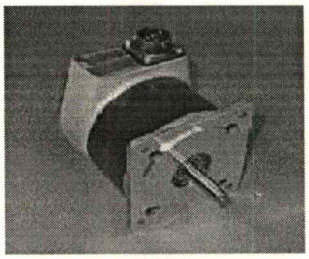
มอเตอร์แบบเป็นขั้น (Stepping Motor) เป็นมอเตอร์ที่ทำงานแบบต่อเนื่อง โดยการแปลงคลื่นสัญญาณที่ป้อนให้เป็นการเคลื่อนที่เชิงมุม ความเที่ยงตรงของระบบจะขึ้นกับความสามารถของมอเตอร์ในการแบ่งขั้นการหมุนตามจำนวนคลื่นของสัญญาณที่ป้อนเข้าระบบ ในมอเตอร์แบบนี้แรงบิดจะลดลงเมื่อความเร็วในการหมุนเพิ่มขึ้น ดังนั้นจะเหมาะสำหรับเครื่องจักรกลเล็กๆ เช่น พร็อตเตอร์

มอเตอร์กระแสสลับ (AC Motor) มอเตอร์ชนิดนี้จะใช้วงจรควบคุมที่ยุ่งยากซับซ้อนกว่า 2 ประเภทข้างต้น แต่ในขนาดที่เท่ากัน จะมีแรงบิดสูงกว่า จะใช้กับงานที่ต้องการแรงบิดที่สูงและขนาดของมอเตอร์จะไม่ใหญ่นักด้วย

รูปที่ 2.4 แสดงมอเตอร์ไฟฟ้าแบบต่างๆ







เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

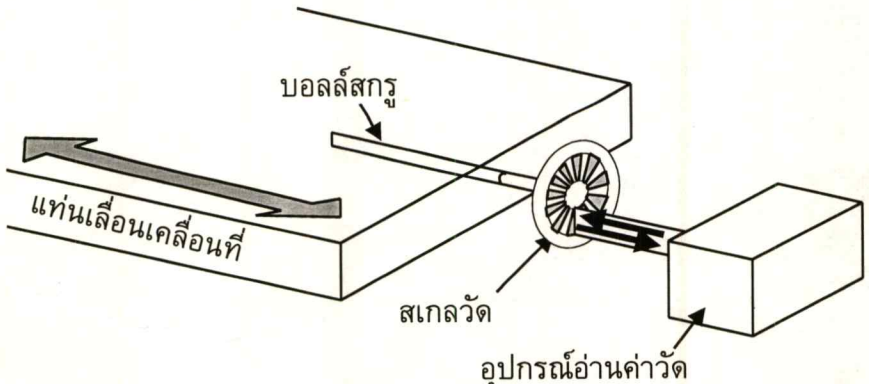
ระบบวัดการเคลื่อนที่ ในแต่ละแนวแกนการเคลื่อนที่ของแท่นไปยังตำแหน่งต่างๆ จะถูกส่งกลับไปยังระบบควบคุมโดยระบบวัดขนาด การวัดการเคลื่อนที่ของแท่นเลื่อนสามารถที่จะวัดได้โดยทั้งทางตรง (Direc Measurement) และทางอ้อม (Indirect Measurement)

-การวัดตำแหน่งโดยตรง จะใช้สเกลวัดยึดติดกับแท่นเลื่อน หรือ โต๊ะงานโดยตรงดังรูปที่ 2.5 การวัดแบบนี้มีข้อดีอยู่ที่ สามารถจำกัดความไม่เที่ยงตรงของสกรูนำเลื่อน และระบบขับจะไม่มีผลการทบต่อค่าที่อ่านได้ โดยอุปกรณ์อ่านค่าวัดจะอ่านข้อมูลในการวัดจากขีดสเกลวัด และ แปลงข้อมูลนี้เป็นสัญญาณไฟฟ้าส่งกลับไปยังระบบควบคุม

รูปที่ 2.5 แสดงการวัดตำแหน่งโดยตรง

การวัดตำแหน่งทางอ้อม ในแบบนี้จะใช้แผ่นจานสัญญาณ(pulse disc) ไปติดกับแกนของบอลล์สกรู และ เมื่อบอลล์สกรูหมุน ก็จะทำให้อุปกรณ์เปลี่ยนค่าวัดอ่านค่าการเคลื่อนที่หมุนของแผ่นจาน ระบบควบคุมก็จะใช้สัญญาณที่ได้รับนี้ไปทำการคำนวณระยะทางการเคลื่อนที่ของแท่นเลื่อนจากอัตราการหมุน (rotation pulses) ของแผ่นจานสัญญาณดังรูปที่ 2.6

รูปที่ 2.6 แสดงการวัดตำแหน่งทางอ้อม

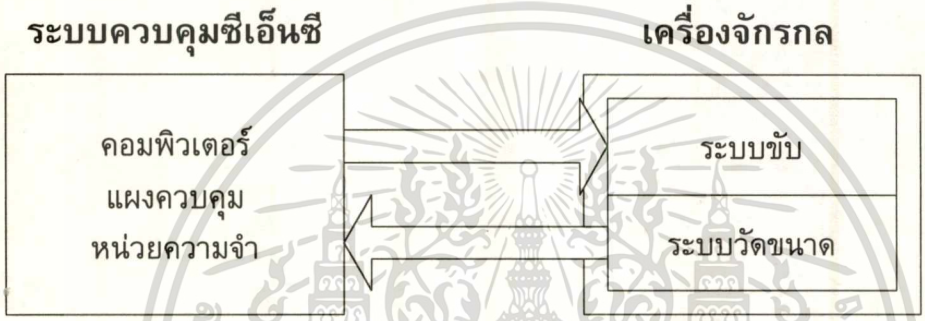


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.2 ส่วนควบคุม

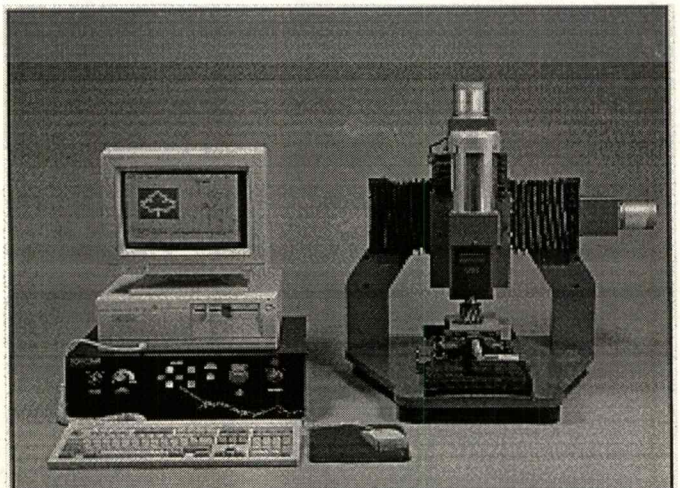
ในระบบซีเอ็นซีจะมีคอมพิวเตอร์อยู่ในระบบ ซึ่งเป็นตัวเก็บข้อมูลโปรแกรมต่างๆ และยังเป็นตัวประมวลผลควบคุมเครื่องจักรกลอีกด้วย ทำให้ช่างควบคุมเครื่องสามารถเขียนโปรแกรมได้ด้วยตัวเอง และ ยังสามารถแก้ไขโปรแกรมหลังจากป้อนเข้าไปในระบบควบคุมเครื่องได้

รูปที่ 2.8 แสดงระบบควบคุมเครื่องจักรกลซีเอ็นซี



ในส่วนของการควบคุมในเครื่องซีเอ็นซีจะแบ่งเป็น 2 ส่วน คือ การควบคุมการเคลื่อนที่ของแนวแกนในลักษณะต่างๆ ให้เครื่องมือเคลื่อนที่ตามรูปทรงเรขาคณิตของชิ้นงาน และควบคุมหน้าที่การทำงานของเครื่องจักรกล (Control of machine function) เช่น การควบคุมระบบหล่อเย็น การเปิดสัญญาณไฟแสดงการทำงานเครื่อง เป็นต้น

รูปที่ 2.9 แสดงการใช้คอมพิวเตอร์ส่วนบุคคลควบคุมเครื่องจักรกลซีเอ็นซี



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

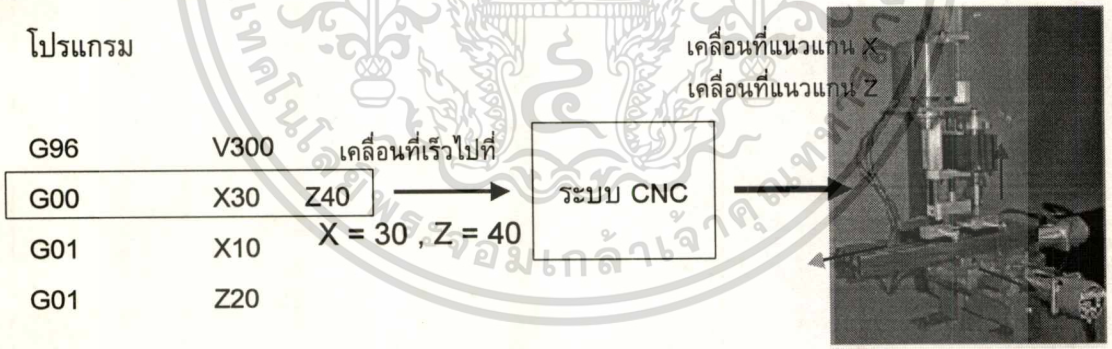
2.2.3 ส่วนโปรแกรมสั่งงานเครื่องจักร

ระบบควบคุมเครื่องจักรกลซีเอ็นซี จะเป็นส่วนที่ทำหน้าที่ควบคุมให้เครื่องจักรกลทำงานตามขั้นตอนต่างๆ ที่กำหนดไว้ การที่เครื่องจักรกลเหล่านั้นจะทำงานได้ จะต้องมีโปรแกรม ที่ช่างผู้ควบคุมเครื่อง หรือ ช่างเขียนโปรแกรมป้อนเข้าไปในระบบควบคุม และเรียกโปรแกรมที่ป้อนเข้าในระบบว่า โปรแกรมเอ็นซี จากนั้นระบบควบคุมจะอ่านโปรแกรมเอ็นซีและเปลี่ยนข้อมูลเหล่านี้ให้เป็นสัญญาณควบคุมสำหรับเครื่องจักรกล การสร้างโปรแกรมซีเอ็นซีจะมีรูปแบบที่ถูกกำหนดไว้เป็นมาตราฐานไว้แล้ว

โปรแกรมเอ็นซีจะเป็นการกำหนดขั้นตอนการตัดเฉือนทั้งหมด ต่างๆ ของเครื่องจักรไว้พร้อมด้วยเงื่อนไขการทำงานอื่นๆ เช่น อัตราป้อน ความเร็วรอบของเพลางาน เป็นต้น หลังจากที่โปรแกรมเอ็นซีนี้ถูกป้อนเข้าไปในระบบควบคุมแล้ว จะสามารถที่จะนำมาใช้ในการตัดเฉือนงานได้หลายๆ ครั้งตามที่ต้องการ

ถ้าเมื่อไหร่ที่มีความต้องการปรับปรุงเปลี่ยนแปลงแก้ไขโปรแกรมให้เหมาะสม ในระบบควบคุมซีเอ็นซีก็จะสามารถทำได้ด้วยช่างควบคุมเครื่องโดยตรง

รูปที่ 2.10 แสดงผลของคำสั่งในโปรแกรมเอ็นซี



ในรูปที่ 2.10 คำสั่งโปรแกรมเอ็นซีเขียนว่า เคลื่อนที่เร็วไปยังจุด X=30, Z=40 จะทำให้มอเตอร์ขับในแกน X และแกน Z เริ่มทำงาน จะทำไปจนกว่าจะเคลื่อนที่ไปยังจุด X=30,Z=40

โปรแกรมเอ็นซีจะประกอบด้วยคำสั่งเป็นบรรทัดต่อๆ กัน เรียก แต่ละบรรทัดว่า บล็อก (Block) ซึ่งเป็นการเขียนขึ้นตามขั้นตอนการตัดเฉือนที่กำหนดไว้รวมกันนั่นเอง

โปรแกรมบล็อกจะแบ่งออกได้โดยการใช้หมายเลขบล็อก เช่น N10 N20 N30 เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาษาโปรแกรมเอ็นซี จะเป็นบล็อกที่ประกอบด้วยคำ(Words) หลายๆ คำรวมกัน คำเหล่านี้จะประกอบขึ้นจากตัวอักษร หรือ สัญลักษณ์กับตัวเลขรวมกัน

รูป 2.11 แสดงโปรแกรมบล็อกที่ประกอบด้วยคำ 7 คำ

คำที่ใช้ในโปรแกรมบล็อกอาจจะทำหน้าที่เป็นคำสั่ง หรือ เป็นเงื่อนไขเสริมสำหรับการทำงานก็ได้ ขึ้นอยู่กับตัวอักษรและตัวเลขที่กำกับอยู่ ตัวอักษรคำสั่งที่มีความสำคัญมาก คือ G คำสั่ง G (G00-G99) ส่วนมากจะเป็นคำสั่งที่ใช้เกี่ยวกับการควบคุมการเคลื่อนที่ของเครื่องมือ

ส่วนอักษรที่ใช้สำหรับเงื่อนไขเสริมที่สำคัญได้แก่

X,Y,Z : ข้อมูลโคออร์ดิเนท
F : อัตราป้อน
S : ความเร็วรอบของเพลางาน

ในการเขียนภาษาโปรแกรมสำหรับระบบควบคุมซีเอ็นซี ผู้ผลิดจะระบุ

-คำสั่งที่สามารถโปรแกรมได้มีอะไรบ้าง

-การทำงานเสริมด้านใดบ้างที่สามารถใช้ร่วมกับคำสั่งแต่ละคำ

-ตัวอักษร และ ตัวเลขใดบ้างที่ใช้เขียนเป็นคำสั่ง และการทำงานเสริม

เมื่อโปรแกรมควบคุมได้ถูกป้อนเข้าไปในระบบควบคุมแล้ว ระบบควบคุมจะทำการตรวจสอบว่าการเขียนโปรแกรมนั้นเป็นไปตามกฎหรือไม่ ส่วนการป้อนข้อมูลโคออร์ดิเนทที่ไม่ถูกต้องโดยช่างเขียนโปรแกรมจะตรวจสอบพบได้เมื่อโปรแกรมทำงานแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.4 ชุดคำสั่งควบคุมเครื่องจักรที่ใช้ในโครงงาน

ชุดคำสั่ง G (G-Code)

ชุดคำสั่ง G-Code ที่ใช้ในโครงงานเป็นชุดคำสั่งที่พื้นฐานบางส่วน โดยนำมาจากเครื่อง Fanuc มีรายละเอียดดังนี้

ตารางที่ 2.1 แสดงชุดคำสั่งที่มีใช้ในโปรแกรม

คำสั่ง	ความหมาย
G00	เข้าสู่ตำแหน่งที่กำหนดอย่างรวดเร็ว
G01	กัดงานเป็นเส้นตรง
G02	กัดงานเป็นแนวเส้นโค้งตามเข็มนาฬิกา CW
G03	กัดงานเป็นแนวเส้นโค้งทวนเข็มนาฬิกา CWW
G04	หยุดชั่วขณะหนึ่ง
G20	ป้อนหน่วยเป็นนิ้ว
G21	ป้อนหน่วยเป็นมิลลิเมตร
G90	โปรแกรมแบบคิดระยะจุดแรกไปถึงจุดสุดท้าย (Absolute)
G91	โปรแกรมแบบคิดระยะจุดหนึ่งไปถึงอีกจุดหนึ่ง (Incremental)

รายละเอียดเกี่ยวกับ G-Code

1 คำสั่ง G00 เป็นการเคลื่อนที่เข้าสู่ตำแหน่งอย่างรวดเร็ว

รูปแบบ G00 X___ Y___ Z____

2 คำสั่ง G01 เป็นการตัดตามระยะทางที่เป็นเส้นตรง เคลื่อนที่จากจุดหนึ่งไปยังจุดที่กำหนดด้วยความเร็วตาม Feed Rate ซึ่งการกำหนดตำแหน่งจะขึ้นกับว่าเป็น G91 หรือ G90

รูปแบบ G01 X___ Y____

หรือ G01 X___ Z ____ F____

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3 คำสั่ง G02,G03 ตัดตามแนวเส้นโค้งตามเข็มนาฬิกา และทวนเข็มนาฬิกา CW .CCW โดยที่ คำสั่ง G02 เป็นการเคลื่อนที่ตามเข็มนาฬิกา และ G03 เป็นการเคลื่อนที่ทวนเข็มนาฬิกา

รูปแบบ G02 X___ Y____ Z____ I____ J____ K____ F___
G03 X___ Y____ Z____ I____ J____ K____ F___

โดยที่ X,Y เป็นจุดปลายของการเคลื่อนที่ และ ค่า I,J เป็นจุดศูนย์กลางของการเคลื่อนที่

4 คำสั่ง G04 เป็นการสั่งให้หยุดชั่วขณะหนึ่งตามเวลา P(หน่วยเป็น วินาที)

รูปแบบ G04 P____ (0,001 sec - 999.999 sec)

5 คำสั่ง G20 เป็นการกำหนดการป้อนหน่วยเป็นนิ้ว (Input in Inch)

รูปแบบ G20

6 คำสั่ง G21 เป็นการกำหนดการป้อนหน่วยเป็นมิลลิเมตร (Input in mm.)

รูปแบบ G21

7 คำสั่ง G90 เป็นการกำหนดการคิดระยะทางแบบเทียบกับจุดอ้างอิง

รูปแบบ G90

8 คำสั่ง G91 เป็นการกำหนดการคิดระยะทางแบบคิดระยะจุดต่อจุด

รูปแบบ G91

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชุดคำสั่ง M (M Function)

ชุดคำสั่ง M ใช้ควบคุม Cutter หรือ Spindle ชุดคำสั่งที่จัดทำขึ้นในโครงงานได้แก่

ตารางที่ 2.2 แสดงชุดคำสั่ง M (M Function)

คำสั่ง	ความหมาย
M02	จบโปรแกรม
M03	ให้ Cutter หมุนตามเข็มนาฬิกา
M04	ให้ Cutter หมุนทวนเข็มนาฬิกา
M05	ให้ Cutter หยุดหมุน

นอกจากชุดคำสั่ง G และ M แล้วยังมีคำสั่งย่อยซึ่งใช้รวมกับชุดคำสั่ง G และ M ได้แก่

คำสั่ง F (Feed Function)

เป็นคำสั่งกำหนดความเร็วของการกัด (cutting Feed Rate) ซึ่งค่าป้อนจะเป็น inch/minute หรือ mm./minute

คำสั่ง S (Spindle Speed Function)

เป็นคำสั่งกำหนดความเร็วของการหมุน Spindle มีหน่วนเป็น rpm.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 การทำงานของเครื่องซีเอ็นซี

ลักษณะการควบคุมการเคลื่อนที่ของแท่นเลื่อนต่างๆ ในเครื่องจักรกลซีเอ็นซีมีอยู่ 2 ลักษณะ คือ

การเคลื่อนที่ในแนวเส้นตรง (Linear Interpolation) การเคลื่อนที่ในลักษณะนี้ คอมพิวเตอร์ในระบบจะทำการคำนวณหาตำแหน่งของจุดต่างๆ ที่ต่อกันในแนวเส้นตรงระหว่างตำแหน่งของเครื่องมือ 2 ตำแหน่ง ในขณะที่เครื่องมือเคลื่อนที่จากจุดหนึ่งไปยังอีกจุดหนึ่งนั้น ระบบควบคุมซีเอ็นซีจะตรวจสอบ และแก้ไขแนวแกนในการเคลื่อนที่ให้ถูกต้องอยู่ตลอดเวลา ทำให้การเคลื่อนที่ของเครื่องมือไม่ผิดพลาด หรือ คลาดเคลื่อนออกจากจุดต่อของเส้นตรงมากกว่าค่าพิกัดความเผื่อของเครื่องที่กำหนดไว้

การเคลื่อนที่แนวเส้นโค้ง (Circular interpolation) ระบบควบคุมซีเอ็นซีจะคำนวณหาตำแหน่งของจุดต่างๆ ที่ต่อกันเป็นเส้นโค้งตามขนาดรัศมีที่กำหนดระหว่างตำแหน่งของเครื่องมือที่กำหนดไว้ 2 ตำแหน่ง ระบบควบคุมจะอาศัยจุดเหล่านี้ในการตรวจสอบ และแก้ไขแนวการเคลื่อนที่ของเครื่องมือให้ถูกต้อง และอยู่ภายในพิกัดความเผื่อของเครื่องจักรกลที่กำหนด

2.3.1 ระบบควบคุมการเคลื่อนที่ของเครื่องมือ

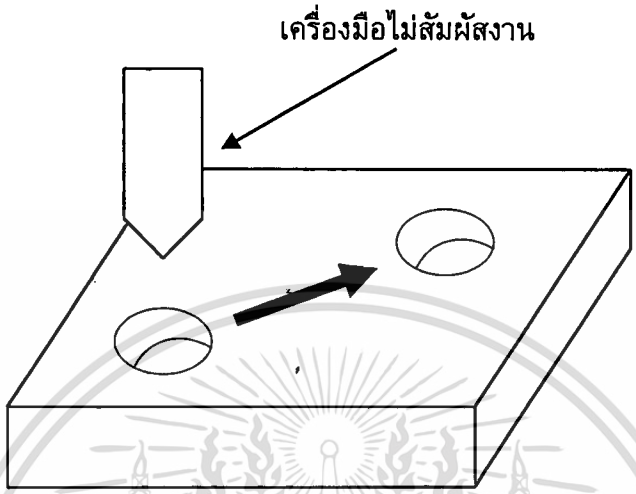
ในระบบควบคุมซีเอ็นซีจะแบ่งการควบคุมการเคลื่อนที่ทั้งสองลักษณะตามลักษณะการเคลื่อนที่ป้อนออกเป็น 3 ชนิด คือ

2.3.1.1 การควบคุมการเคลื่อนที่แบบจุดต่อจุด

การควบคุมแบบนี้จะควบคุมการเคลื่อนที่ของเครื่องมือระหว่างจุดสองจุดที่โปรแกรมไว้ในลักษณะการเคลื่อนที่เร็ว (Rapid traverse) โดยที่เครื่องมือจะต้องไม่สัมผัสชิ้นงาน ดังรูป 2.12 แนวแกนในการเคลื่อนที่ขึ้นอยู่กับชนิดของระบบควบคุม คือ มอเตอร์ขับระบบป้อนอาจเริ่มทำงานหลายๆ แนวแกนพร้อมๆ กัน หรือ ทำงานทีละแนวแกน จนกว่าจะเคลื่อนที่ถึงตำแหน่งของเครื่องมือที่โปรแกรมไว้ การควบคุมแบบจุดต่อจุดมักจะใช้กับเครื่องเจาะ เครื่องเชื่อมจุด เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

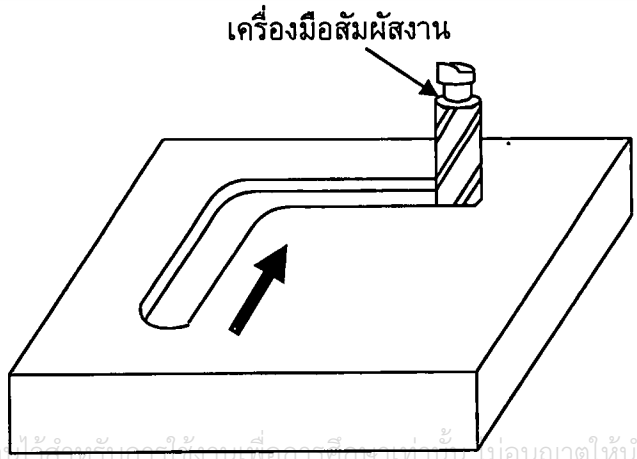
รูปที่ 2.12 แสดงการควบคุมแบบจุดต่อจุด



2.3.1.2 การควบคุมการเคลื่อนที่แบบตัดตรง

การควบคุมการเคลื่อนที่ชนิดนี้ สามารถใช้ควบคุมการเคลื่อนที่แบบเคลื่อนที่เร็ว และยังสามารถใช้ควบคุมการเคลื่อนที่ของเครื่องมือในแนวขนานกับแนวแกนของเครื่องจักรกล ตามค่าอัตราป้อนที่ต้องการ แต่จะสามารถควบคุมการเคลื่อนที่ได้ครั้งละ 1 แนวแกนเท่านั้น การเคลื่อนที่ของเครื่องมือจะถูกควบคุมด้วยอัตราป้อน และความยาวในการเคลื่อนที่ดังรูป 2.13 ระบบการควบคุมการตัดเฉือนแนวเส้นตรงชนิดนี้จะใช้กับเครื่องกัด และเครื่องกลึงแบบง่ายๆ

รูปที่ 2.13 แสดงการควบคุมแบบตัดตรง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.1.3 การควบคุมการเคลื่อนที่ที่เป็นเส้นโค้ง หรือ ตามเส้นขอบรูป

การควบคุมแบบนี้สามารถควบคุมการเคลื่อนที่ทำงานได้ดังนี้

-ควบคุมเครื่องมือให้เคลื่อนที่ไปยังตำแหน่งที่ต้องการแบบเคลื่อนที่เร็วได้

-ควบคุมเรื่องมือให้เคลื่อนที่ขนานกับแนวแกนไปยัง ตำแหน่งที่ต้องการตามค่าอัตราป้อนได้

-ควบคุมเครื่องมือให้เคลื่อนที่ไปยังตำแหน่งใดๆ บนชิ้นงานที่กำหนดในแนวเส้นตรงและเส้นโค้งตามค่าอัตราป้อนได้

รูปที่ 2.14 แสดงการควบคุมตามเส้นขอบรูป

การควบคุมตามเส้นขอบรูปนี้ยังแบ่งได้อีกเป็น 3 ระดับ ขึ้นกับความสามารถของระบบควบคุม คือ ความสามารถในการควมคุมการเคลื่อนที่ของเครื่องมือได้ 2 หรือ 3 แกนพร้อมๆกัน ระดับการควบคุมทั้ง 3 ระดับเป็นดังนี้

-ควบคุมตามเส้นขอบรูปแบบ 2 แกน ระบบควบคุมจะสามารถควบคุมเครื่องมือให้เคลื่อนที่ในระนาบ(plane) ที่กำหนดเฉพาะได้ 2 แนวแกนพร้อมๆ กัน ทำให้สามารถเคลื่อนที่ได้ทั้งแนวเส้นตรง และเส้นโค้ง แต่จะไม่สามารถเปลี่ยนระนาบในการทำงานได้

-การควบคุมตามเส้นขอบรูปแบบ 2 แกนครึ่ง ระบบควบคุมนี้จะควบคุมเครื่องมือให้เคลื่อนที่ในแนวเส้นตรง และเส้นโค้งบนระนาบใดๆ ที่ต้องการก็ได้ แต่จะไม่สามารถควบคุมการเคลื่อนที่ได้ใน 2 แนวแกนพร้อมกันเท่านั้น

-การควบคุมตามเส้นขอบรูปแบบ 3 แกน ระบบควบคุมจะสามารถควบคุมเครื่องมือให้เคลื่อนที่ในแนวเส้นตรง และเส้นโค้งได้พร้อมๆกันทั้ง 3 แนวแกน เป็นลักษณะ 3 มิติได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

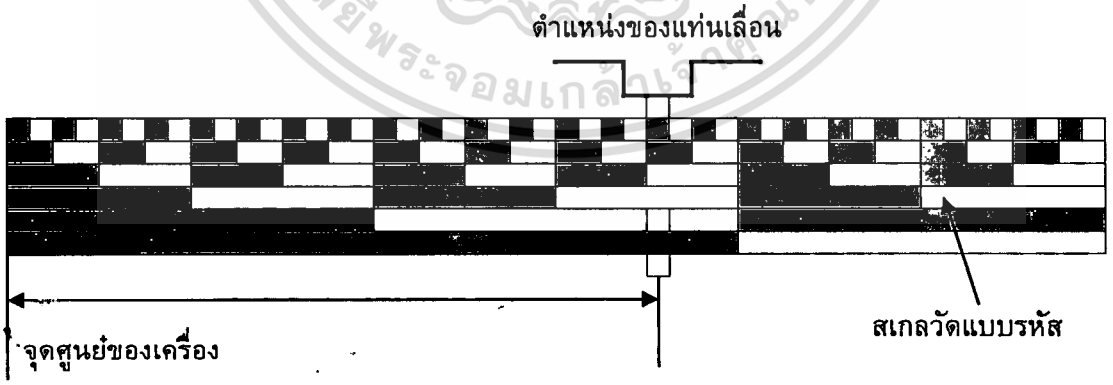
2.3.2 การวัดระยะการเคลื่อนที่ตำแหน่งของเครื่องมือ

ในระบบซีเอ็นซีการวัดระยะการเคลื่อนที่ของตำแหน่งเครื่องมือ ต้องการการวัดการเคลื่อนที่ที่มีความเที่ยงตรงตลอดแนวแกนป้อน จะต้องต่อระบบขับป้อนเข้ากับอุปกรณ์วัดที่เหมาะสม อุปกรณ์วัดจะประกอบด้วยสเกล กับอุปกรณ์อ่านค่าที่อ่านค่าจากสเกลได้ สเกลที่ใช้ในอุปกรณ์การวัดมีอยู่ 2 ชนิดด้วยกัน คือ สเกลแบบรหัส (code measuring scale) และ สเกลแบบช่อง (division grid) การใช้สเกลทั้ง 2 ชนิดนี้จะขึ้นอยู่กับวิธีวัดตำแหน่ง (position measurement) วิธีการวัดตำแหน่งที่ใช้กันโดยทั่วไปมี 2 วิธี คือ การวัดคำแหน่งแบบสัมบูรณ์ (absolute position measurement) กับ การวัดตำแหน่งอย่างต่อเนื่อง หรือ แบบลูกโซ่ (incremental or chain position measurement) มีรายละเอียดดังนี้

2.3.2.1 การวัดตำแหน่งสัมบูรณ์

การวัดตำแหน่งแบบสัมบูรณ์ เป็นการวัดค่าตำแหน่งต่างๆ ที่สามารถวัดค่าได้ตลอดเวลา เป็นอิสระจากสถานะการทำงานของเครื่อง และระบบควบคุม และค่าที่วัดได้นี้จะอ้างอิงจากจุดศูนย์อ้างอิง (fixed zero datum) เสมอ

รูปที่ 2.15 แสดงการใช้สเกลวัดตำแหน่งแบบสัมบูรณ์

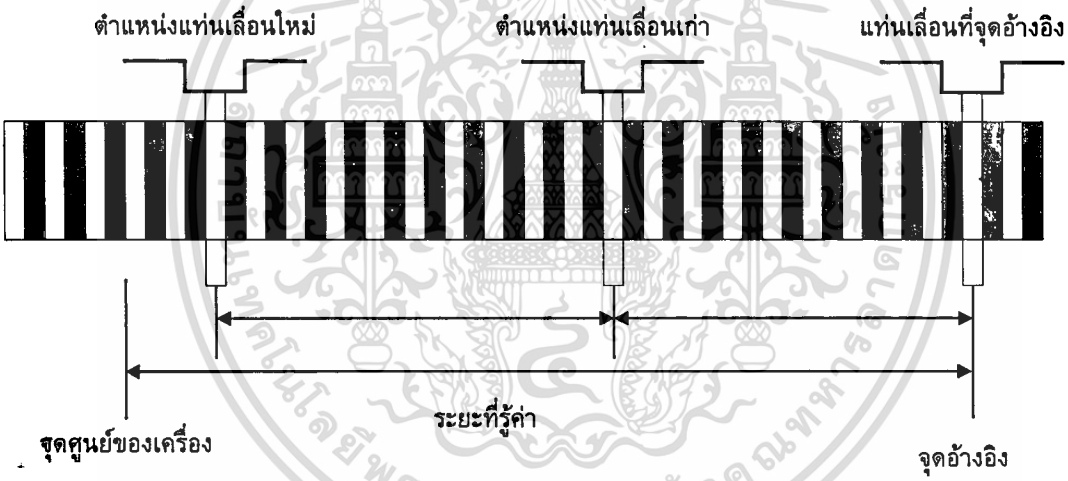


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.2.2 การกำหนดตำแหน่งส่วนเพิ่ม

เป็นการวัดตำแหน่งแบบต่อเนื่อง ตัวสเกลวัดจะแบ่งเป็นช่อง (grid) อย่างง่าย โดยแต่ละช่องจะมีพื้นที่มืดสว่างสลับกันไป ในการวัดตำแหน่งเป็นการเพิ่ม หรือ ลดขนาดความยาวในการเคลื่อนที่ จากตำแหน่งที่วัดอยู่ในระหว่างการเคลื่อนที่ของแท่นเลื่อน ระบบควบคุมจะทำการนับจำนวนช่องแบ่งที่แท่นเลื่อนได้เคลื่อนที่ไป แล้วนำค่าไปคำนวณหาตำแหน่งใหม่ที่แตกต่างจากตำแหน่งเดิม เป็นตำแหน่งสุดท้ายของแท่นเลื่อนเสมอ

รูปที่ 2.16 แสดงสเกลการวัดตำแหน่งแบบต่อเนื่อง

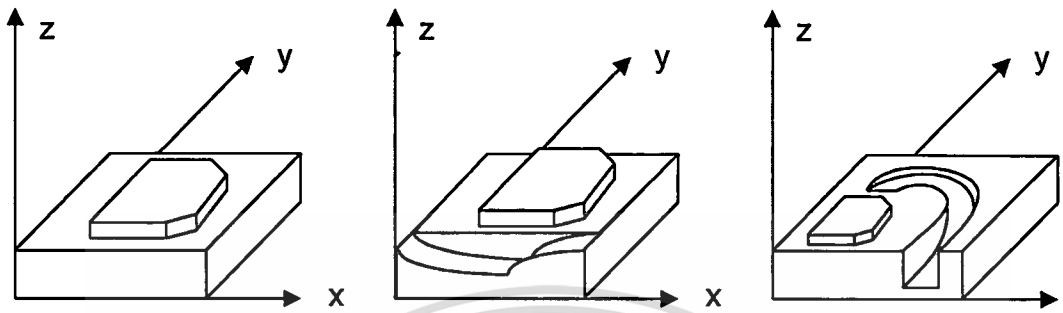


2.3.3 รูปแบบการทำงานของเครื่องมือ

การทำงานของเครื่องมือสามารถ ทำงานแบบ 2 มิติ แบบ 2 มิติ และ แบบ 3 มิติได้ ดังแสดงในรูป 2.17 การทำงานแบบ 2 มิติ คือ การสร้างชิ้นงานให้อยู่ในระนาบเดียว มีการควมคุมการทำงานเพียง 2 แกน ซึ่งโดยทั่งไปจะอยู่ในระนาบ XY ส่วนแกน Z เป็นการกำหนดค่าที่แน่นอน เช่น ความลึก สำหรับการทำงานแบบ 2 มิติ จะคล้ายกับแบบ 2 มิติ แต่เพิ่มในส่วนแกน Z ที่มีค่าเปลี่ยนแปลงแต่ไม่เปลี่ยนแปลงพร้อมกันทั้ง 3 แกน ส่วนการทำงานแบบ 3 มิติ จะเป็นการสร้างชิ้นงานโดยที่แกนทั้ง 3 ทำงานพร้อมกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.17 แสดงลักษณะการสร้างชิ้นงาน



ก) แบบ 2 มิติ ข) แบบ 2 1/2มิติ ค) แบบ 3 มิติ

ด้วยลักษณะการสร้างชิ้นงานดังที่กล่าวมาแล้ว นอกจากสามารถนำไปใช้ได้กับงานกัดยังสามารถนำไปประยุกต์ใช้กับงานเจาะซึ่งถือว่าเป็นการสร้างชิ้นงานแบบ 2 มิติ เมื่อความลึกในการเจาะคงที่ ถ้าความลึกไม่คงที่ก็จะเป็นงานแบบ 2 มิติ ส่วนงานเดินขอบสามารถสร้างชิ้นงานได้ทุกรูปแบบ ส่วนงานกัดเป็นแอ่งหลุมจะเป็นการสร้างชิ้นงานแบบ 2 มิติ

2.3.4 ขั้นตอนการสร้างชิ้นงาน

การใช้งานเครื่องซีเอ็นซี จะต้องคำนึงถึงขั้นตอนของการทำงานเพื่อให้ได้งานที่มีคุณภาพ ดังนั้นจึงต้องมีการออกแบบขั้นตอนการใช้งาน ดังต่อไปนี้

ก) ขั้นตอนการวางแผน ขั้นตอนนี้ทำหลังจากที่ผู้ออกแบบได้ออกแบบมาแล้ว เป็นการกำหนดว่าต้องทำส่วนไหนก่อนส่วนไหนหลัง เพื่อให้ได้งานตามที่ออกแบบไว้

ข) ขั้นตอนการเขียนโปรแกรมสั่งงาน เป็นการเขียนคำสั่งนำมาเรียงลำดับขั้นตอนการทำงานตามที่ได้ออกแบบไว้

ค) ขั้นตอนการตรวจสอบชุดคำสั่ง ในขั้นตอนนี้ทำได้ 2 วิธี คือ วิธีที่หนึ่ง ทำการทดสอบการทำงานกับเครื่องซีเอ็นซี ได้โดยใช้วัสดุอื่นทดสอบ วิธีที่สองคือ ใช้โปรแกรมจำลองการทำงาน (Simulation) ในเครื่องคอมพิวเตอร์ทำการทดสอบก่อนทำงานจริง เพื่อเป็นการป้องกันความผิดพลาดที่อาจเกิดขึ้นได้

ง) ขั้นตอนการผลิต หมายถึง ตั้งแต่การเตรียมชิ้นงาน เตรียมเครื่องมือและอุปกรณ์ต่างๆ เพื่อความพร้อมที่จะทำงานได้ ซึ่งหลังจากเดินเครื่องแล้ว เครื่องซีเอ็นซีจะทำงานโดยอัตโนมัติตามขั้นตอนการสั่งงานที่ได้ออกแบบไว้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
การออกแบบและสร้างส่วนควบคุม

3.1 บทนำ

ในการออกแบบควบคุมการทำงานของเครื่องต้นแบบที่สร้างขึ้นในโครงงานนี้จะทำการแยกส่วนการควบคุมมอเตอร์ออกเป็น 3 ส่วนหลักๆคือ

1. ส่วนควบคุมมอเตอร์
2. ส่วนของการติดต่อกับ PLC
3. ส่วนติดต่อกับคอมพิวเตอร์

3.2 ส่วนของระบบการควบมอเตอร์ของเครื่องต้นแบบ

การออกแบบในส่วนนี้จะใช้ เซอร์โวมอเตอร์เป็นตัวต้นกำลังการขับเคลื่อนแกน X แกน Y แกน Z และหัวหมุน (Spindle) ซึ่งจะประกอบด้วยอุปกรณ์ที่สำคัญต่างๆ คือ

3.2.1 การ์ดควบคุมเซอร์โวมอเตอร์

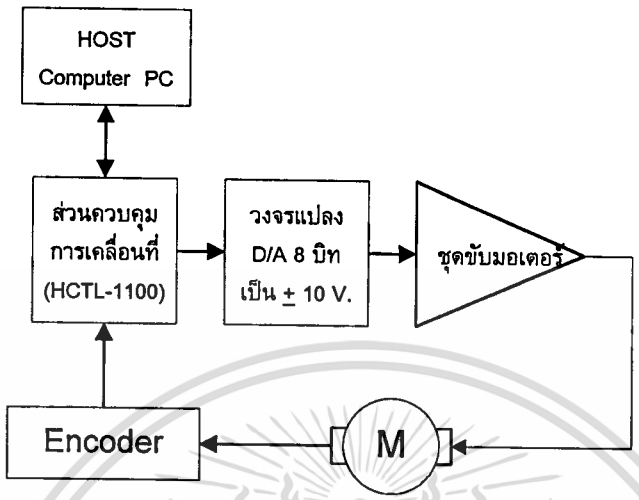
การ์ดควบคุมเซอร์โวมอเตอร์ที่ออกแบบขึ้นซึ่งมีคุณลักษณะและคุณสมบัติพื้นฐานดังนี้

- สามารถควบคุมตำแหน่งและความเร็วการหมุนเดินหน้าและถอยหลังได้
- สามารถตั้งค่าคำสั่ง (Command) การหมุนจาก Host Prcessor by PC
- มีพอร์ทรับค่าการเข้าระหัส (Encoder) จากตำแหน่งการหมุนของแกนโรเตอร์ของมอเตอร์เพื่อที่จะนำค่าดังกล่าวไปตรวจสอบตำแหน่งการการหมุนมอเตอร์

ดังนั้นเพื่อให้ตรงตามความต้องการเบื้องต้นการออกแบบจึงใช้วงจรรวม (IC) สำเร็จรูปที่เหมาะสมเบอร์ HCTL-1100 ซึ่งมีลักษณะการควบคุมใช้งานดังรูปที่ 3.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.1 แสดงลักษณะการควบคุมของ HCTL-1100



ซึ่ง HCTL-1100 มีคุณสมบัติและหน้าที่การใช้งานที่จำเป็นดังต่อไปนี้คือ

1) เป็นไอซีที่มีระบบการประมวลผลทางดิจิตอล (Digital Signal Processing ; DSP)

2) มีพอร์ทสำหรับต่อ Encoder หรือเซ็นเซอร์ประเภทอื่น เช่น ฮอลล์เซ็นเซอร์

3) สามารถควบคุมตำแหน่งและความเร็วได้โดยการเซ็ตป้อนค่าคำสั่งควบคุมมอเตอร์(Motor Command)

4) มีพอร์ทขนานขนาด 8 บิตเอาต์พุต สำหรับการสร้างเป็นสัญญาณที่ใช้ควบคุมมอเตอร์

5) สามารถใช้เป็นตัวควบคุมมอเตอร์ได้หลายชนิดและรวมถึง DC Motor ที่มีแปรงถ่านหรือไม่มีแปรงถ่าน อีกทั้งสามารถใช้ได้กับ Step Motor

6) มีโหมดควบคุมการทำงานอยู่ 4 แบบคือ

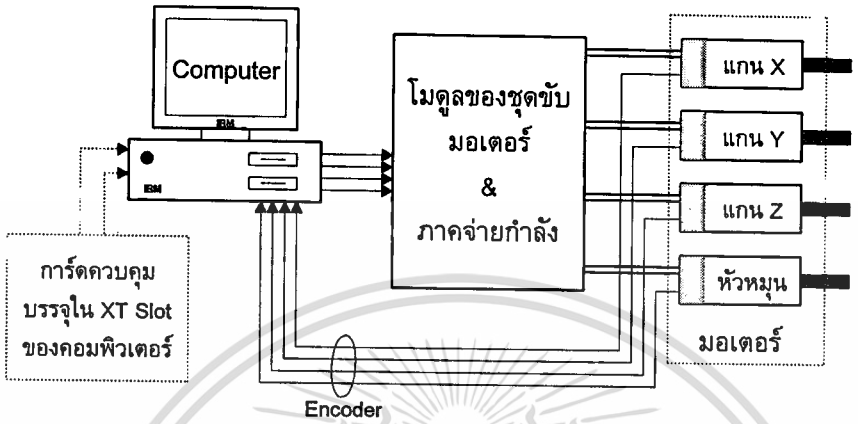
- แบบควบคุมตำแหน่ง (Position Control)
- แบบควบคุมความเร็ว (Proportional Velocity)
- แบบควบคุมความเร่ง (Integral Velocity)
- แบบตำแหน่งและความเร็วแบบสี่เหลี่ยมคางหมู (Trapezoid Profile)

7) มี 8 บิต I/O command พอร์ทให้ทำงานตามโหมดการทำงานทั้ง 4 แบบ

ในการออกแบบทางฮาร์ดแวร์ของการ์ดที่ใช้ควบคุมดังกล่าว ใช้ HCTL-1100 จำนวน 4 ตัวโดยที่แต่ละตัวจะแยกการควบคุมการขับเคลื่อนแกน X แกน Y แกน Z และหัวหมุนออกจากกัน การทำงานจะขึ้นอยู่กับการเขียนซอฟท์แวร์หรือส่วนของโปรแกรมที่ผนวก เข้ากับส่วนที่เป็นฮาร์ดแวร์ของการ์ดควบคุมเพื่อที่จัดลำดับการทำงานของมอเตอร์แต่ละตัวให้มีความสัมพันธ์กันตามที่ได้แสดงไว้ในรูปที่ 3.2 ซึ่งแสดงระบบควบคุมมอเตอร์ของการ์ดควบคุม

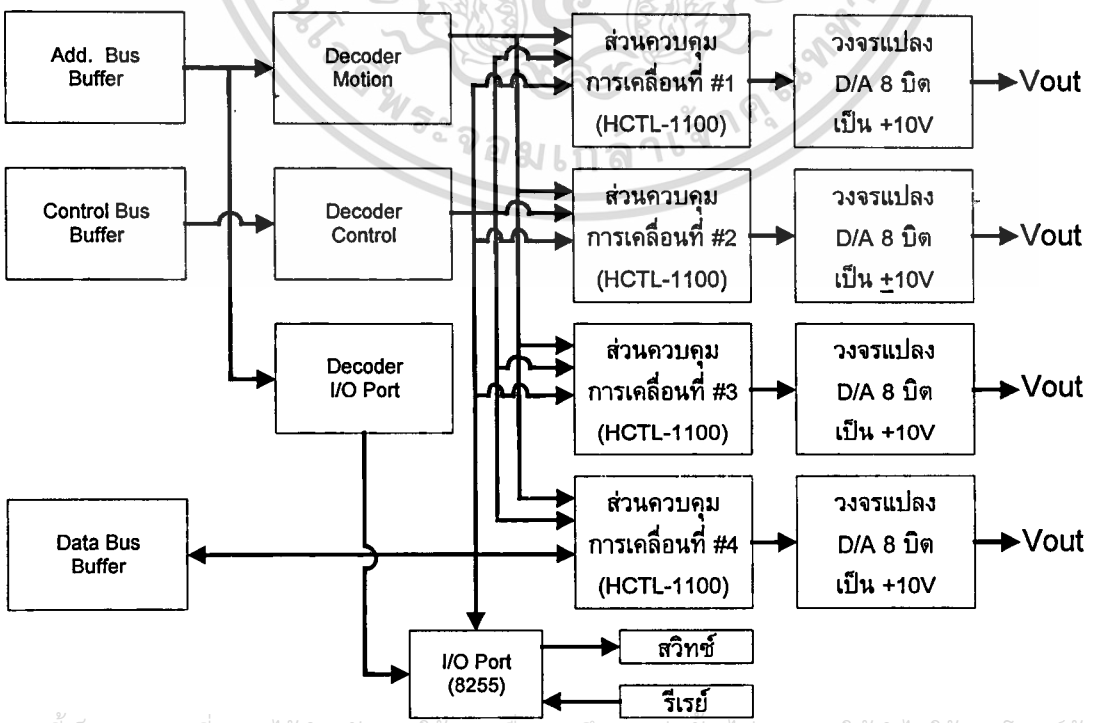
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.2 แสดงระบบควบคุมมอเตอร์ของการ์ดควบคุม



จากรูปที่ 3.2 ได้ออกแบบในส่วนของบอร์ดควบคุมให้สามารถควบคุมมอเตอร์ได้ 4 ตัว และมีพอร์ทสำหรับอินพุท เอาต์พุตอีกจำนวนหนึ่งโดยมีบล็อกไดอะแกรมของบอร์ดควบคุมเป็นดังรูปที่ 3.3

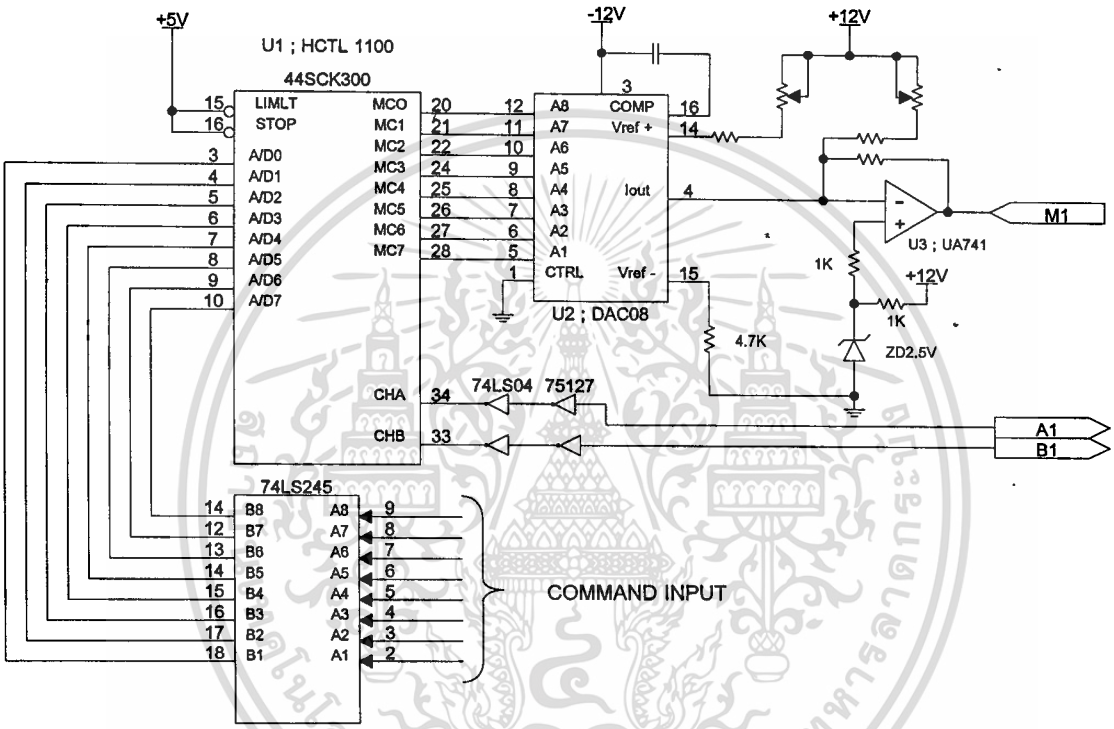
รูปที่ 3.3 แสดงบล็อกไดอะแกรมของการ์ด



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการออกแบบวงจรภายในการ์ดควบคุมที่ใช้ HCTL-1100 ที่ใช้ในเครื่องต้นแบบ แสดงในรูปที่ 3.4 ซึ่งในการควบคุมมอเตอร์ทุกๆ ตัวก็จะมีวงจรในลักษณะเดียวกัน

รูปที่ 3.4 แสดงวงจรย่อยภายในการ์ดควบคุมที่ใช้ HCTL-1100



จากรูปที่ 3.4 เมื่อ 74LS245 ทำหน้าที่เป็นวงจร Buffer รับค่า Command เข้าทาง A1-A8 ขนาด 8 บิต ออกทาง B1-B8 ส่งไปยังขาอินพุตของ HCTL-1100 คือ AD0-AD8 ซึ่งสัญญาณดังกล่าว จะไปประมวลผลเป็นค่าข้อมูล และออกทางขา MC0 (LSB) - MC7 (MSB) ค่าที่ได้นำไปแปลงเป็น สัญญาณทางอะนาลอกโดยใช้ DAC08 แต่สัญญาณที่ออกมาจะเป็นสัญญาณในรูปของกระแส แล้วจึงทำการเปลี่ยนเป็นค่าของระดับแรงดันโดยใช้ออปแอมป์เพื่อให้ได้ระดับแรงดันที่ต้องการคือ -10 V ถึง +10 V เป็นค่าระดับแรงดันที่ได้นำไปควบคุมหน่วยขับมอเตอร์อีกต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

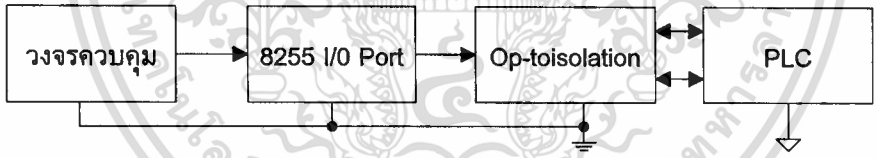
3.3 ส่วนของการติดต่อกับ PLC

จะใช้ 8255 เป็นตัวขยายพอร์ทอินพุตและเอาต์พุตและทำการแยกกราวด์ระหว่างวงจรควบคุม และวงจรของเครื่อง PLC โดยที่ 8255 สามารถที่จะโปรแกรมให้เป็นอินพุต หรือ เอาต์พุตได้ตามที่ต้องการโดยในตัว 8255 มี I/O พอร์ทขนาด 8 บิต อยู่ 3 พอร์ท กับอีก 1 พอร์ทควบคุม ที่ใช้ติดต่อกับวงจรควบคุมที่ต่อเข้ากับวงจรถอดระหัส I/O พอร์ท เป็นตัวเลือกหรือกำหนดติดต่อกับ I/O พอร์ท คือ A,B,C นอกจากนี้ยังมีสัญญาณควบคุมอีก 4 เส้น คือ

- RD เป็นสัญญาณในการอ่านพอร์ทของ 8255
- WR เป็นสัญญาณในการเขียนข้อมูลมาที่ พอร์ทของ 8255
- CS เป็นขาสัญญาณเปิดให้ 8255 ทำงาน
- Reset เป็นการเริ่มต้นการทำงานของ 8255 ใหม่

การออกแบบในการใช้ 8255 และวงจรป้องกันจะใช้การแยกระบบกราวด์ทางไฟฟ้าออกจากกันอย่างสิ้นเชิง โดยการใช้การเชื่อมโยงทางแสงเข้าช่วยหรือเรียกทั่วๆไปว่า Op-to Isolation ซึ่งจะแสดงตามบล๊อกไดอะแกรมดังนี้คือ

รูปที่ 3.5 แสดงบล๊อกไดอะแกรมของการติดต่อกับ PLC

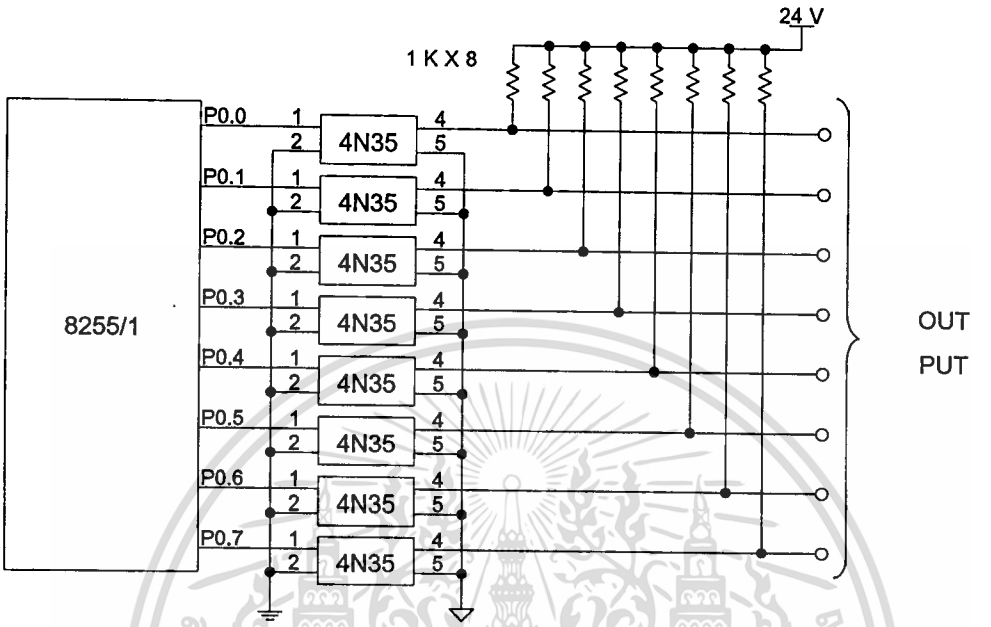


3.3.1 การเชื่อมต่อวงจรด้านเอ้าพุตของการ์ดกับอุปกรณ์ PLC

ในการแยกกราวด์ดังกล่าวจะเป็นการป้องกันวงจรควบคุมในกรณีที่เกิดการทำงานผิดพลาดจากวงจรของ PLC และการออกแบบจะใช้ 8255 จำนวน 2 ตัว มาทำหน้าที่รับส่งค่าจาก PLC ที่ระดับแรงดันของวงจรควบคุม 5 โวลท์ และระดับแรงดันใช้งานของ PLC ที่ 24 โวลท์ โดยที่จะแยกการรับค่ากับการส่งค่าคนละตัว การขยายพอร์ทนี้สามารถขยายได้ถึง 48 บิต คือ แยกเป็น 24 บิตของ 8255 ตัวแรกใช้เป็นอินพุตอย่างเดียว 8255 ตัวที่สองใช้เป็นเอาต์พุตอย่างเดียว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.6 แสดงการต่อ 8255 กับ Op-to ทางด้านเอาต์พุต



จากรูปที่ 3.6 แสดงการต่อวงจรตัวอย่างของ 8255 เพียง 8 บิตเอาต์พุตและใช้ Op-to เบอร์ 4N35 เป็นตัวเชื่อมต่อวงจรควบคุมกับวงจรของ PLC

ในที่นี้จะใช้งานเบื้องต้น คือ ชุดรีเรย์สำหรับอินเอเบิลชุดขับมอเตอร์แต่ละแกน และ ชุดขับหัวหมุน ดังนี้

-Enable Axis X สำหรับอนุญาติให้แกน X เคลื่อนที่ได้

-Enable Axis Y สำหรับอนุญาติให้แกน Y เคลื่อนที่ได้

-Enable Axis Z สำหรับอนุญาติให้แกน Z เคลื่อนที่ได้

-Enable Spindle สำหรับอนุญาติให้หัวหมุนหมุน

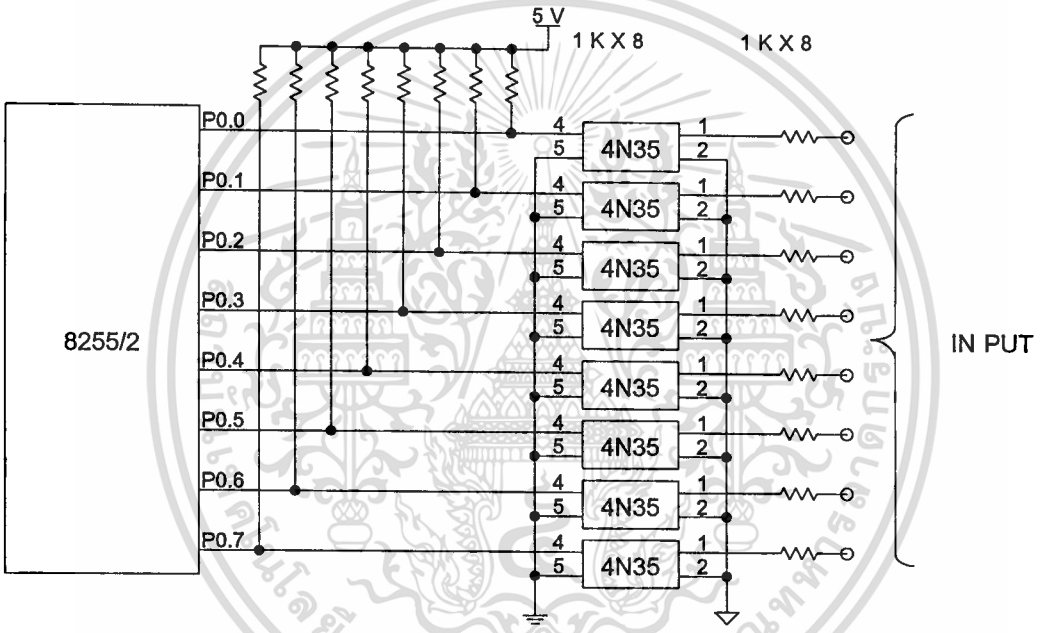
ในส่วนเอ้าพุทที่เหลือก็เป็นการจัดเตรียมไว้สำหรับงานอื่นๆ เช่น สัญญาณไฟเตือน ประกอบการเคลื่อนที่ การต่ออุปกรณ์ชุดหล่อเย็น เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.2 การเชื่อมต่อวงจรควบคุมกับ PLC ทางอินพุตของ 8255

ในทางกลับกันถ้าเราต้องการรับสัญญาณที่ส่งมาจากการทำงานของ PLC ในการตรวจสอบสถานะทางลอจิกที่จะใช้เป็นค่าอ้างอิงการทำงานของชุดควบคุม เราจะใช้ 8255 ตัวที่ 2 เป็นตัวรับค่าหรือข้อมูลที่มาจากเอาต์พุตพอร์ทของ PLC ซึ่งจะแสดงในรูปที่ 3.7

รูปที่ 3.7 แสดงการต่อ 8255 กับ Op-To ทางด้านอินพุท



การใช้งานอินพุทในเบื้องต้นนี้จะมีดังนี้

ชุดสวิทซ์สำหรับหาตำแหน่งเริ่มต้นของการเคลื่อนที่ของแต่ละแกน(Zero Set)

-Home Switch X สวิทซ์สำหรับหาตำแหน่งเริ่มต้นแกน X

-Home Switch Y สวิทซ์สำหรับหาตำแหน่งเริ่มต้นแกน Y

-Home Switch Z สวิทซ์สำหรับหาตำแหน่งเริ่มต้นแกน Z

ชุดสวิทซ์สำหรับกำหนดขอบเขตของการเคลื่อนที่สำหรับแต่ละแกน

-Limit Switch X สวิทซ์สำหรับกำหนดขอบเขตแกน X

-Limit Switch Y สวิทซ์สำหรับกำหนดขอบเขตแกน Y

-Limit Switch Z สวิทซ์สำหรับกำหนดขอบเขตแกน Z

ส่วนสวิทซ์อินพุทที่เหลือมีไว้สำหรับอินพุทอื่นๆ ตามแต่ลักษณะการใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 ส่วนติดต่อกับคอมพิวเตอร์

เครื่องคอมพิวเตอร์ที่ใช้เป็นแบบ IBM PC ที่มีช่องผ่านข้อมูลแบบ XT ซึ่งเป็นช่องผ่านข้อมูลขนาด 8 บิท ที่อยู่บริเวณหลังเครื่องคอมพิวเตอร์ เพื่อเป็นตัวเชื่อมระบบคอมพิวเตอร์เข้ากับระบบควบคุมการเคลื่อนที่ในแต่ละแกน และเชื่อมเข้ากับสวิทซ์ รีเรย์ ที่ควบคุมเครื่องจักรอีกทีหนึ่ง ในการเชื่อมต่อมีรายละเอียดดังนี้ การดีโค้ดพอร์ทสำหรับใช้งานโดยใช้ไอซี 74Ls244 จำนวน 2 ตัว และ 74Ls245 เป็นบัฟเฟอร์แอดเดรสบัส และบัฟเฟอร์ดาต้าบัส และสำหรับการดีโค้ดใช้ 74Ls688 และ 74Ls154 เป็นจำนวน 2 ตัว สำหรับดีโค้ดพอร์ทได้ถึง 32 พอร์ท 300-30F เป็นพอร์ทของ 74Ls154 ตัวที่หนึ่ง และ 310-31F เป็นพอร์ทของ 74Ls154 อีกตัวหนึ่ง โดยพอร์ทต่างๆ จะถูกใช้งานดังในตารางที่ 3.1

ตารางที่ 3.1 แสดงการใช้งานพอร์ทต่างๆ ที่เชื่อมต่อเข้ากับคอมพิวเตอร์

ตำแหน่งพอร์ท	หน้าที่
300-302H	สัญญาณควบคุมหน่วยควบคุมแกนที่ 1 (แกน X)
303-305H	สัญญาณควบคุมหน่วยควบคุมแกนที่ 2 (แกน Y)
306-308H	สัญญาณควบคุมหน่วยควบคุมแกนที่ 3 (แกน Z)
309-30BH	สัญญาณควบคุมหน่ายควบคุมการหมุนของหัวหมุน
30C-30EH	สำหรับอินพุทข้อมูลจากเอ็นโค้ดเดอร์อีกตัว
310-311H	สำหรับต่อ 8255 ตัวที่ 1 เป็นอินพุทพอร์ท สำหรับต่อกับสวิทซ์ต่างๆ
312-313H	สำหรับต่อ 8255 ตัวที่ 2 เป็นเอ้าพุทพอร์ท สำหรับต่อกับรีเรย์ต่างๆ (สำหรับสวิทซ์ และ รีเรย์)

ความหมายของพอร์ทต่างๆ เป็นดังนี้

1 พอร์ท 300 เป็นสัญญาณ ALE-0 สำหรับเอ็นเอเบิลชิพควบคุมมอเตอร์แกนที่ 1

2 พอร์ท 301 เป็นสัญญาณ OE-0 สำหรับสัญญาณเอ้าพุทเอ็นเอเบิลชิพควบคุมมอเตอร์ตัวที่ 1

3 พอร์ท 302 เป็นสัญญาณ CS-0 สำหรับสัญญาณชิพซีเล็คสำหรับชิพควบคุมมอเตอร์ตัวที่ 1

4 พอร์ท 303 เป็นสัญญาณ ALE-1 สำหรับเอ็นเอเบิลชิพควบคุมมอเตอร์แกนที่ 2

5 พอร์ท 304 เป็นสัญญาณ OE-1 สำหรับสัญญาณเอ้าพุทเอ็นเอเบิลชิพควบคุมมอเตอร์ตัวที่ 2

6 พอร์ท 305 เป็นสัญญาณ CS-1 สำหรับสัญญาณชิพซีเล็คสำหรับชิพควบคุมมอเตอร์ตัวที่ 2

7 พอร์ท 306 เป็นสัญญาณ ALE-2 สำหรับเอ็นเอเบิลชิพควบคุมมอเตอร์แกนที่ 3

8 พอร์ท 307 เป็นสัญญาณ OE-2 สำหรับสัญญาณเอาต์พุตเอ็นเอเบิลชิพควบคุมมอเตอร์ตัวที่ 3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

9 พอร์ท 308 เป็นสัญญาณ CS-2 สำหรับสัญญาณชิพซีเล็คสำหรับชิพควบคุมมอเตอร์ตัวที่ 3
10 พอร์ท 309 เป็นสัญญาณ ALE-3 สำหรับเอ็นเอเบิลชิพควบคุมมอเตอร์แกนที่ 4
11 พอร์ท 30A เป็นสัญญาณ OE-3 สำหรับสัญญาณเอ้าพุทเอ็นเอเบิลชิพควบคุมมอเตอร์ตัวที่ 4
12 พอร์ท 30B เป็นสัญญาณ CS-3 สำหรับสัญญาณชิพซีเล็คสำหรับชิพควบคุมมอเตอร์ตัวที่ 4
13 พอร์ท 310-311H เป็นขาสัญญาณ A0,A1 สำหรับ 8255 ตัวที่ 1 ที่ใช้เป็นอินพุทพอร์ท
14 พอร์ท 312-313H เป็นขาสัญญาณ A0,A1 สำหรับ 8255 ตัวที่ 2 ที่ใช้เป็นเอาต์พุตพอร์ท

3.5 สรุปในส่วนของฮาร์ดแวร์

จากฮาร์ดแวร์ที่ออกแบบไว้นี้จะสามารถควบคุมมอเตอร์ในแต่ละตัวได้อย่างอิสระ ดังเราไม่เพียงแต่เราจะใช้ควบคุมเครื่องจักรซีเอ็นซีแบบเครื่องกัดแนวดิ่งได้เท่านั้น เรายังสามารถใช้ในการควบคุมเครื่องจักรอื่นๆ ที่ใช้มอเตอร์ในการควบคุมและมีมอเตอร์ไม่เกิน 4 ตัวได้ ในส่วนของการรับอินพุท และ ส่งเอาต์พุตก็สามารถเพิ่มเติมได้อีกมาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
การออกแบบและสร้างในส่วนของโปรแกรม

4.1 ภาพรวมของโปรแกรมที่ออกแบบ

ในการออกแบบโปรแกรมสำหรับควบคุมเครื่องกัดแนวดิ่งซีเอ็นซีนั้น เราได้ออกแบบโดยใช้ภาษาซีในการพัฒนาโปรแกรม ซึ่งตัวโปรแกรมที่ออกแบบจะแบ่งเป็นส่วนสำคัญๆ ดังนี้

-โปรแกรมส่วนใช้เชื่อมต่อระหว่างคอมพิวเตอร์ กับบอร์ดควบคุมมอเตอร์ที่สร้างขึ้น

-โปรแกรมส่วนติดต่อกับผู้ใช้งานในลักษณะของการเขียน และ แก้ไขโปรแกรมสั่งงานและควบคุมเครื่องซีเอ็นซี

-โปรแกรมในส่วนการติดต่อกับข้อมูลภายนอก ในรูปของไฟล์ข้อมูลจากโปรแกรมวาดแบบที่ใช้รูปแบบไฟล์ข้อมูลแบบ dxf

-โปรแกรมส่วนที่ใช้แสดงภาพจำลองการทำงาน และแสดงภาพ 3 มิติของชิ้นงาน

4.1.1 โปรแกรมส่วนควบคุมมอเตอร์

ในส่วนของการควบคุมมอเตอร์จะแบ่งตามลักษณะของโปรแกรมได้เป็น 2 ลักษณะ คือ

-ส่วนที่ใช้สั่งงานให้กับมอเตอร์ในแต่ละแนวแกน เช่น สั่งให้หมุน หรือ หยุด กำหนดทิศทาง ความเร็ว และความเร่ง เป็นต้น

-ส่วนที่ใช้สำหรับสั่งให้มอเตอร์เคลื่อนที่อย่างสัมพันธ์กันในแต่ละแกนให้เป็นรูปทรงที่ต้องการ

โดยจะมีรายละเอียดของฟังก์ชั่นใช้งานในแต่ละส่วนดังนี้

1 โปรแกรมย่อยที่เป็นฟังก์ชั่นสำหรับสั่งงานให้มอเตอร์หมุน หรือ หยุดในลักษณะต่างๆ ในส่วนนี้เป็นชุดคำสั่งสำหรับควบคุมมอเตอร์ในแต่ละแกน แบ่งตามลักษณะการใช้งานได้ ดังนี้

1 ควบคุมแกน(Axis Control) มีฟังก์ชั่นในส่วนนี้ดังนี้

set_1() เปลี่ยนแกนปัจจุบันไปยังแกนที่ 1 คำสั่ง set_1() จะเปลี่ยนแกนปัจจุบันไปยังแกน 1 คำสั่งทั้งหมดที่ตามมาภายหลังจะมีผลต่อแกน 1 ทั้งหมด

set_2() เปลี่ยนแกนปัจจุบันไปยังแกนที่ 2 คำสั่ง set_2() จะเปลี่ยนแกนปัจจุบันไปยังแกน 2 คำสั่งทั้งหมดที่ตามมาภายหลังจะมีผลต่อแกน 2 ทั้งหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

set_3() เปลี่ยนแกนปัจจุบันไปยังแกนที่ 3 คำสั่ง set_3() จะเปลี่ยนแกนปัจจุบันไปยังแกน 3 คำสั่งทั้งหมดที่ตามมาภายหลังจะมีผลต่อแกน 3 ทั้งหมด

set_4() เปลี่ยนแกนปัจจุบันไปยังแกนที่ 4 คำสั่ง set_4() จะเปลี่ยนแกนปัจจุบันไปยังแกน 4 คำสั่งทั้งหมดที่ตามมาภายหลังจะมีผลต่อแกน 4 ทั้งหมด

2 ควบคุมรูปแบบการเคลื่อนที่

set_prel_trap() เปลี่ยนรูปแบบการเคลื่อนที่เป็นแบบ trapezoidal point-to-point ในโหมดนี้จะต้องมีการระบุตำแหน่งที่จะเคลื่อนที่ไป คำสั่ง set_pos() ความเร็วสูงสุด คำสั่ง set_vel() และความเร่ง คำสั่ง set_acc() ที่ใช้ โดยสามารถจะทำการเปลี่ยนค่าตำแหน่งและความเร็วได้ตลอดเวลาแม้ในขณะที่มีการเคลื่อนที่ แต่ความเร่งจะเปลี่ยนไม่ได้แกนใดที่ถูกกำหนดให้อยู่ในโหมดนี้ ก็จะอยู่ไปจนกว่าจะเปลี่ยนโหมดใหม่

set_prel_vel() เปลี่ยนรูปแบบการเคลื่อนที่เป็นแบบ velocity contouring ในโหมดนี้จะต้องมีการกำหนดความเร่ง คำสั่ง set_acc() และความเร็วสูงสุด คำสั่ง set_vel() ที่ใช้ โดยสามารถทำการเปลี่ยนค่าความเร่ง และความเร็วสูงสุดได้ตลอดเวลา แม้ในขณะกำลังเคลื่อนที่ แกนใดที่ถูกกำหนดให้เคลื่อนที่ในโหมดนี้จะทำงานในโหมดนี้ตลอดไป จนกว่าจะมีการกำหนดโหมดการเคลื่อนที่ใหม่

set_pos() กำหนดค่าตำแหน่ง เป็นการกำหนดตำแหน่งปลายทางที่จะเคลื่อนที่ไป ค่าตำแหน่งที่กำหนดจะยังไม่ถูกนำไปใช้จนกว่าจะใช้คำสั่ง update()

set_vel() กำหนดค่าความเร็ว เป็นการกำหนดค่าความเร็วสูงสุดที่จะมีได้ counts/sample มีค่าอยู่ระหว่าง 0 ถึง 16383 ค่าความเร็วที่กำหนดให้จะถูกนำมาใช้ก็ต่อเมื่อใช้คำสั่ง update() ก่อน

set_acc() กำหนดความเร่ง จะเป็นการกำหนดขนาดความเร่งที่ใช้ ซึ่งอยู่ในรูปจำนวน count/sample/asmple ค่าความเร่งที่ตั้งไว้จะถูกใช้เมื่อมีคำสั่ง update() คำสั่งนี้ใช้ได้ในโหมด trapezoidal point to point หรือ velocity contouring เท่านั้น

clr_prfl() กำหนดค่าความเร็ว ความเร่ง และตำแหน่งให้เป็น 0 ฟังก์ชั่นนี้จะถูกใช้เมื่อมีคำสั่ง update() คำสั่งนี้ยังใช้ในการหยุดมอเตอร์กะทันหันด้วย

synch_prfl() เป็นการกำหนดให้ค่า target position ให้มีค่าเท่ากับ actual position ในขณะนั้น คำสั่งนี้ใช้ได้กับการเคลื่อนที่ทุกแบบ มักใช้ในขณะเกิดความผิดพลาดในการเคลื่อนที่เพื่อที่จะทำให้ระยะเป้าหมายที่ตั้งไว้กับระยะที่เคลื่อนที่ไปจริงมีค่าเท่ากันอีกครั้ง ฟังก์ชั่นนี้จะทำงานเมื่อเรียกคำสั่ง update()

zero_pos() กำหนดค่า actual position และ target position ให้เป็นศูนย์ คำสั่งนี้จะทำเมื่อเรียกคำสั่ง update()

get_pos() อ่านค่าตำแหน่งจากการ์ดควบคุม จะทำการอ่านค่าตำแหน่งปลายทางที่ถูกกำหนดโดยคำสั่ง set_pos() ค่าที่กลับมาจะอยู่ในหน่วยของจำนวน count

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

get_vel() จะทำการอ่านค่าความเร็วที่ถูกกำหนดโดยคำสั่ง set_vel() ค่าที่ถูกส่งกลับมาจะอยู่ในหน่วยของ count/sample ซึ่งจะเป็นเลขจำนวนเต็ม

get_acc() อ่านค่าความเร่งจากการ์ดควบคุมที่ถูกเซ็ตด้วยคำสั่ง set_acc() ค่าความเร่งที่อ่านได้จะอยู่ในหน่วย count/sample/sample ซึ่งจะเป็นเลขจำนวนเต็ม

get_trgt_pos() อ่านค่าตำแหน่งที่จะเคลื่อนที่ไป ค่าที่อ่านได้จะอยู่ในรูป counts

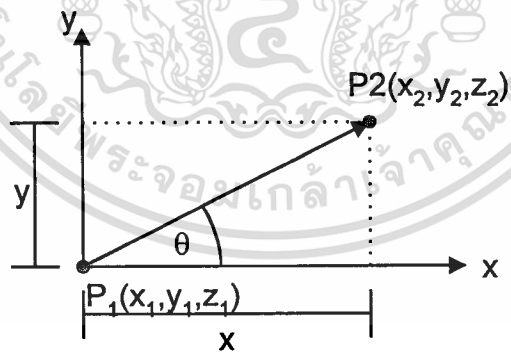
get_actl_pos() อ่านค่าตำแหน่งที่ เคลื่อนที่ไปได้จริงของมอเตอร์ ค่าที่ส่งกลับมาจะอยู่ในหน่วยของ counts เป็นเลขจำนวนเต็ม

get_trgt_vel() เป็นการอ่านค่าความเร็วของแกนใดๆ ที่จะเคลื่อนที่ไปในช่วงเวลาหนึ่งๆ ค่าที่กลับมาอยู่ในหน่วยของ counts/sample ซึ่งเป็นเลขจำนวนเต็ม

2 โปรแกรมย่อยสำหรับการเคลื่อนที่ที่สัมพันธ์กันในแต่ละแกน เป็นการควบคุมการเคลื่อนที่ให้แต่ละแนวแกนเคลื่อนที่สัมพันธ์กันเป็นเส้นตรง หรือ เส้นโค้ง โดยอาศัยทฤษฎีทางคณิตศาสตร์ที่จำเป็นในการคำนวณแก่ยวกับการเคลื่อนที่ ดังนี้

1 การคำนวณตำแหน่งและความเร็วของการเคลื่อนที่เส้นตรง (Linear Interpolation) เป็นการคำนวณสำหรับการเคลื่อนที่เป็นเส้นตรงระหว่างจุดต่อจุดใดๆ ในพิกัด 3 มิติดังรูปที่ 4.1

รูปที่ 4.1 แสดงการกำหนดการเคลื่อนที่เป็นฟังก์ชั่นเส้นตรงเชิงความเร็ว



การเคลื่อนที่เป็นฟังก์ชั่นเส้นตรง ถ้า P1 เป็นจุดเริ่มต้นของการเคลื่อนที่ อยู่ที่พิกัด (X1,Y1,Z1) และ P2 เป็นจุดสิ้นสุดการเคลื่อนที่ อยู่ที่พิกัด (X2,Y2,Z2) ความเร็วในการเคลื่อนที่จาก P1 ไปยัง P2 จะมีค่าเป็น

$$V = \frac{\sqrt{((x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2)}}{t} \qquad (4.1)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ้า R คือ ความยาวของเส้นตรงที่ลากจากจุด P1 มายัง P2 จะได้ว่าความเร็วสัมพัทธ์ของ x, y และ z ที่ทำให้เกิด v หมายถึง การเคลื่อนที่ของคมตัดในเชิงเส้นตรง จะได้ว่า

$$V_x = \frac{V(x_2 - x_1)}{R} \tag{4.2}$$

$$V_y = \frac{V(y_2 - y_1)}{R} \tag{4.3}$$

$$V_z = \frac{V(z_2 - z_1)}{R} \tag{4.4}$$

โดยที่ V = ความเร็วของการเคลื่อนที่ของคมตัด ที่ต้องการกำหนดในการใช้งาน

R = การเคลื่อนที่ของคมตัดในเชิงเส้นตรง

2. การคำนวณตำแหน่งและความเร็วการเคลื่อนที่ในแนวโค้ง (Circular Interpolation)

เป็นการคำนวณสำหรับการเคลื่อนที่เป็นเส้นโค้งจากจุดหนึ่งไปยังอีกจุดหนึ่ง หรือ เป็นวงกลม ข้อมูลที่จำเป็นสำหรับใช้ในการคำนวณเส้นทางเดินของวงกลมได้แก่

-ทิศทางการเคลื่อนที่ว่าตามเข็มนาฬิกา หรือ ทวนเข็มนาฬิกา (CW/CCW)

-จุดเริ่มต้นของส่วนโค้ง

-จุดสิ้นสุดของส่วนโค้ง

-จุดศูนย์กลางของส่วนโค้ง

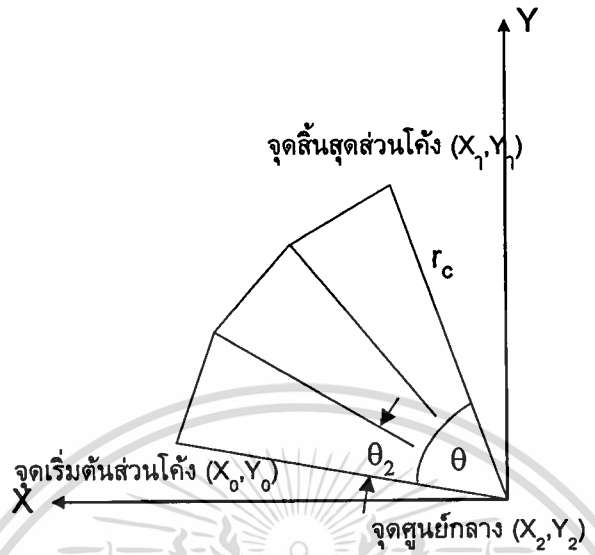
เมื่อได้ข้อมูลที่จำเป็นสำหรับส่วนโค้งมาหมดแล้ว จะนำข้อมูลที่มีอยู่มาทำการคำนวณจุดบนเส้นโค้งด้วยการประมาณค่าเส้นตรง

การประมาณรูปวงกลมด้วยเส้นตรง (Linear approximation of aCircle)

เป็นการแบ่งส่วนของส่วนโค้ง หรือ วงกลม ออกเป็นส่วนเล็กๆ เท่าๆกัน แล้วลากเส้นตรงระหว่างปลายของส่วนโค้งที่แบ่งออกมาเป็นคอร์ด(chard) ที่มีขนาดเท่ากัน โดยที่คอร์ดจะมีความยาวเท่ากับ 0.01 มิลลิเมตร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.2 แสดงการประมาณส่วนของวงกลมด้วยเส้นตรง



จำนวนช่องที่แบ่ง(n) = ความยาวของส่วนโค้ง / ความยาวของคอร์ด

$$n = \frac{\theta * R}{0.01}$$

โดยที่ n จะเป็นเลขจำนวนเต็มที่ปัดเศษขึ้นเสมอ

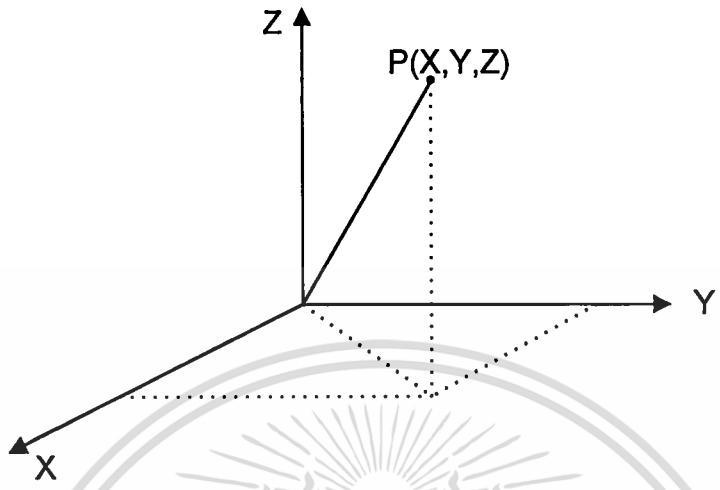
ดังนั้นจะได้ว่าองศาที่เปลี่ยนไปในการเคลื่อนที่แต่ละคอร์ด $\theta_2 = \frac{\theta}{n}$

4.1.2 โปรแกรมส่วนแสดงภาพ 3 มิติของชิ้นงาน

โปรแกรมในส่วนนี้ใช้สำหรับสร้างภาพ 3 มิติของชิ้นงาน จากคำสั่งการควบคุมเครื่องซีเอ็นซีที่เขียนขึ้นมา หลังจากที่ได้ทำการจำลองการทำงานของเครื่องกัดแนวดิ่งซีเอ็นซีแล้ว โดยภาพของวัตถุที่ได้จะอาศัยระดับสีในการแบ่งแยกรูปร่างของวัตถุ ระบบพิกัด 3 มิติที่ใช้ในที่นี้เป็นระบบแบบตั้งฉาก (rectangular coordinate system) ซึ่งจุดในระบบนี้จะถูกกำหนดโดยตัวเลข 3 ตัวในแนวแกน x,y และ z โดยบอกเป็นระยะทางของจุดในแต่ละแกนดังภาพที่ 4.3 ในที่นี้อาศัยทฤษฎีทางคณิตศาสตร์ในการแสดงภาพ 3 มิติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.3 แสดงจุดในระบบพิกัดฉาก 3 มิติ



ทฤษฎีทางคณิตศาสตร์ที่ใช้ในการแสดงภาพ 3 มิติในระบบพิกัดฉาก

-การเคลื่อนย้ายตำแหน่งในระบบพิกัด 3 มิติ การเปลี่ยนแปลงเคลื่อนย้ายจุด (x,y,z) ไปเป็นจุดใหม่ (x',y',z') จะใช้ Tx,Ty,Tz เป็นระยะทางที่เลื่อนจาก x,y,z ตามลำดับ

สมการของการเคลื่อนย้ายจะเป็นดังนี้

$$[x' \quad y' \quad z' \quad 1] = [x \quad y \quad z \quad 1] * \begin{vmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ Tx & Ty & Tz & 1 \end{vmatrix}$$

-การกำหนดขนาดของวัตถุ จะเป็นการกำหนดให้วัตถุในแต่ละแนวแกนใหญ่ขึ้นหรือลดลงตามสัดส่วนที่ต้องการ ในแต่ละแนวแกน โดย Sx,Sy,Sz คือ สัดส่วนของการกำหนดวัตถุในแต่ละแนวแกน จะมีสมการของการกำหนดขนาดดังนี้

$$[x' \quad y' \quad z' \quad 1] = [x \quad y \quad z \quad 1] * \begin{vmatrix} Sx & 0 & 0 & 0 \\ 0 & Sy & 0 & 0 \\ 0 & 0 & Sz & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

-การหมุนรูป 3 มิติ ในการหมุนนี้จะต้องกำหนดเจาะจงว่าจะหมุนในแนวแกนไหน และจะมีสมการสำหรับการหมุนในแต่ละแนวแกนดังนี้

หมุนรอบแกน x เป็นมุม θ_x จะมีสมการในการหมุนดังนี้

$$[x' \quad y' \quad z' \quad 1] = [x \quad y \quad z \quad 1] * R_x(\theta_x$$

โดยที่

$$Rx(\theta) = \begin{vmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\theta) & \sin(\theta) & 0 \\ 0 & -\sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix}$$

หมุนรอบแกน y เป็นมุม θ_y จะมีสมการในการหมุนดังนี้

$$[x' \quad y' \quad z' \quad 1] = [x \quad y \quad z \quad 1] * R_y(\theta_y)$$

โดยที่

$$Ry(\theta) = \begin{vmatrix} \cos(\theta) & 0 & -\sin(\theta) & 0 \\ 0 & 1 & 0 & 0 \\ \sin(\theta) & 0 & \cos(\theta) & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix}$$

หมุนรอบแกน y เป็นมุม θ_z จะมีสมการในการหมุนดังนี้

$$[x' \quad y' \quad z' \quad 1] = [x \quad y \quad z \quad 1] * R_z(\theta_z$$

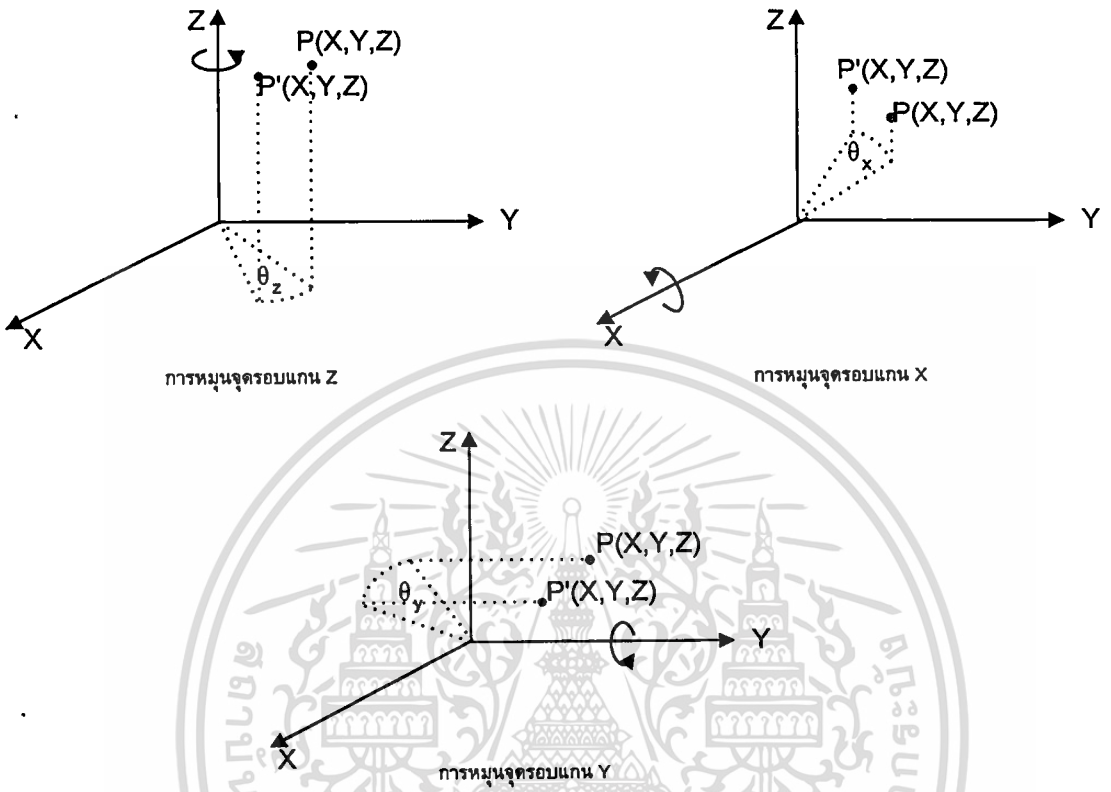
โดยที่

$$Rz(\theta) = \begin{vmatrix} \cos(\theta) & \sin(\theta) & 0 & 0 \\ -\sin(\theta) & \cos(\theta) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix}$$

การหมุนรอบแกน x,y,z ในมุมข้างต้น จะเป็นการหมุนในทิศทางทวนเข็มนาฬิกา ในมุมที่กำหนด

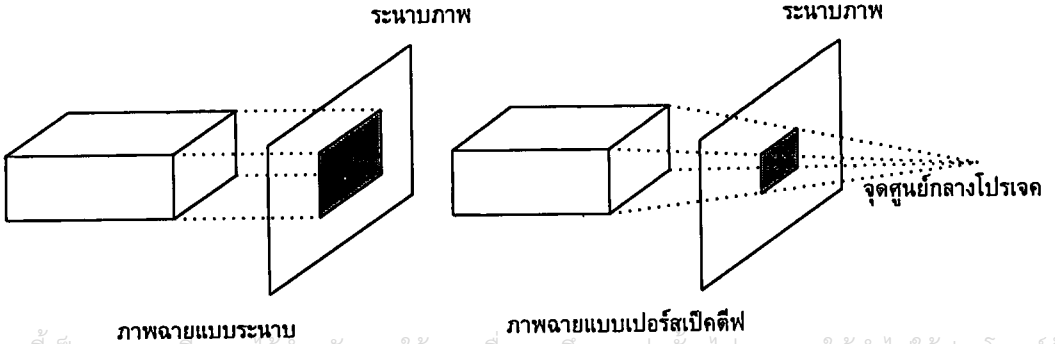
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.4 แสดงการหมุนของจุด และ วัตถุ รอบระบบแนวแกน



การแปลงข้อมูลจากระบบพิกัด 3 มิติ ให้เป็นข้อมูลสำหรับแสดงในระบบพิกัด 2 มิติ ในที่นี้ใช้วิธีฉายเงาของวัตถุให้ตกลงบนระนาบหนึ่ง ที่ระนาบก็จะเกิดเงา หรือ ภาพ 2 มิติของวัตถุนั้น ระนาบนี้เราเรียกว่า ระนาบภาพ (view plane) ดังรูปที่ 4.5 ในวิธีฉายเงานั้นมีหลายประเภท แต่ที่นิยมใช้มีเพียง 2 ประเภท คือ การฉายเงาแบบขนาน (Parallel Projection) และการฉายเงาแบบเปอร์สเป็คทีฟ (Perspective Projection)

รูปที่ 4.5 แสดงการฉายเงาภาพ

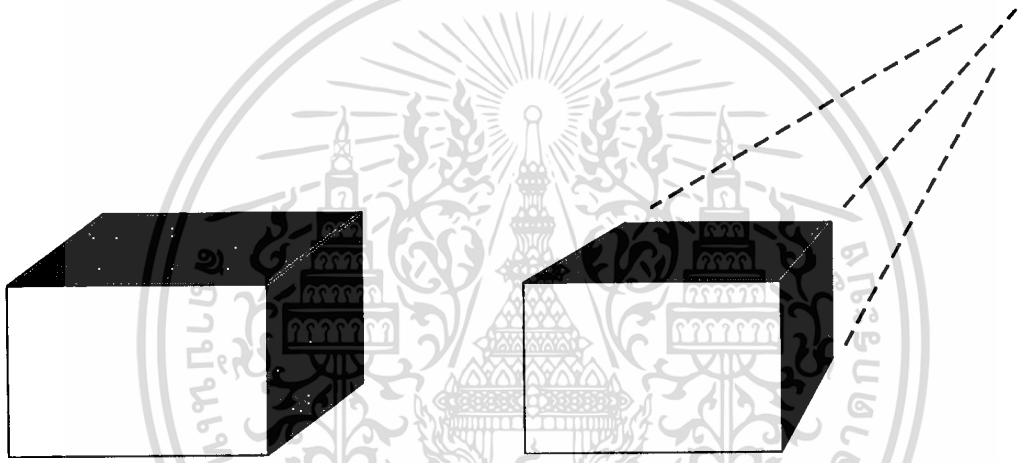


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพฉายเงาแบบขนานนั้นจะมีลักษณะ คือ แสงของเงาที่ฉายจะขนานกันไปหมดตรงสู่ฉากรับภาพ (ในที่นี้คือจอภาพ) ส่วนภาพฉายเงาแบบเปอร์สเป็คทีฟนั้น แสงของเงาทั้งหมดจะพุ่งสู่จุดๆ หนึ่งที่ไม่ขนานกัน จุดที่แสงเงาพุ่งไปนี้เรียกว่า จุดศูนย์กลางโปรเจ็คชั่น (center of projection) ดังรูปที่ 4.5

ลักษณะของภาพที่เกิดขึ้นจากการทำโปรเจคชั่นต่างประเภทกัน จะมีลักษณะต่างกันดังแสดงในรูปที่ 4.6 เป็นการเปรียบเทียบภาพที่ได้จากการทำโปรเจคชั่นแบบขนาน และโปรเจคชั่นแบบเปอร์สเป็คทีฟ

รูปที่ 4.6 แสดงภาพที่ได้จากการทำโปรเจคชั่น



ภาพจากการฉายแบบขนาน ภาพจากการฉายแบบเปอร์สเป็คทีฟ

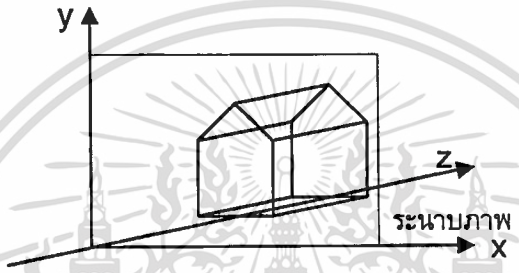
ระนาบภาพ คือ ระนาบที่เราฉายเงาของวัตถุลงไป ภาพของเงาที่เกิดขึ้นบนระนาบจะเป็นภาพ 2 มิติของวัตถุ และเป็นภาพที่จะถูกวาดออกทางจอภาพหรืออุปกรณ์พิมพ์ภาพ ระนาบภาพสามารถเป็นระนาบใดๆ ก็ได้ที่ตั้งอยู่ในโคออร์ดิเนท 3 มิติ ระบบโคออร์ดิเนท 3 มิติที่วัตถุอยู่นี้เรียกว่า ออปเจกโคออร์ดิเนท (Object Coordinate) เราจะมองระนาบภาพนี้เสมือนกับเป็นระนาบ xy ธรรมดาที่วางตัวอยู่ในระบบ มีแกน x แกน y และจุดกำเนิดของระนาบ ดังแสดงในรูป 4.7 ระบบโคออร์ดิเนต XY ของระนาบภาพนี้เรียกว่า โคออร์ดิเนทของระนาบภาพ (View Plane Coordinates)

การวางตำแหน่ง และการวางตัวของภาพต้องใช้พารามิเตอร์อยู่ 4 ตัว เป็นตัวกำหนดดังนี้ พารามิเตอร์ตัวที่ 1 คือ จุดอ้างอิงภาพ(View Reference point) Pr(Xr,Yr,Zr) จุดนี้เป็นจุดศูนย์กลางของความสนใจในการมองภาพ พารามิเตอร์ตัวที่ 2 คือ เวคเตอร์ตั้งฉากของระนาบภาพ (View Plane Normal Vector) N=[Xn Yn Zn] เป็นเวคเตอร์ที่ตั้งฉากกับระนาบภาพ ถ้าเราลากเส้นตรงจากจุดอ้างอิงภาพให้ขนานกับเวคเตอร์ N ไปตั้งฉากกับระนาบภาพ จุดที่เส้นตรง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

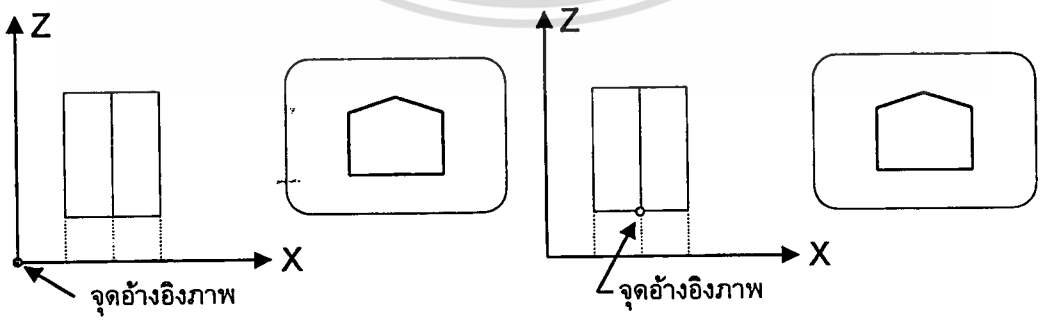
นี้ตัดกับระนาบภาพคือ จุดกำเนิดของโคออร์ดิเนทของระนาบภาพนั่นเอง พารามิเตอร์ตัวที่ 3 คือ ระยะภาพ (View Distance) หมายถึง ระยะห่างระหว่างจุดอ้างอิงภาพ และ ระนาบภาพ โดยปรกติมักกำหนดให้ระยะภาพเป็นศูนย์ ทำให้จุดอ้างอิงภาพ และ จุดกำเนิดของภาพเป็นจุดเดียวกัน พารามิเตอร์ตัวที่ 4 คือ วิวอัฟเวคเตอร์ (View Up Vector) V=[Xv Yv Zv] เป็นเวคเตอร์ที่กำหนดแนวทิศทางตั้งขึ้นของโคออร์ดิเนทของระนาบภาพ นั่นคือ เวคเตอร์ V เป็นเวคเตอร์ที่ขนานและ มีทิศทางเดียวกับแกน Y ของโคออร์ดิเนทของระนาบภาพ

รูปที่ 4.7 แสดงวัตถุในระบบออปเจกโคออร์ดิเนท



ในระบบคอมพิวเตอร์กราฟิค การสร้างภาพต้องกำหนดพารามิเตอร์เหล่านี้ เพื่อให้ได้ลักษณะภาพที่ต้องการ การกำหนดค่าพารามิเตอร์เหล่านี้จะทำให้ได้ภาพที่ต่างกันออกไป การเปลี่ยนจุดอ้างอิงภาพจะทำให้ภาพย้ายตำแหน่งไป ส่วนของวัตถุที่ปรากฏที่จุดกำเนิดบนโคออร์ดิเนทของภาพจะเปลี่ยนไป ดังรูปที่ 4.8 แสดงตัวอย่างผลของการเปลี่ยนจุดอ้างอิงภาพ

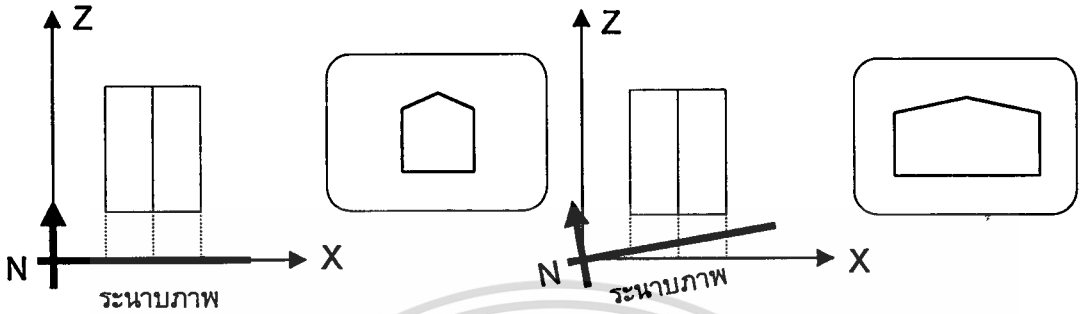
รูปที่ 4.8 แสดงการเปลี่ยนจุดอ้างอิงภาพ



ถ้าเวคเตอร์ตั้งฉากของระนาบภาพเปลี่ยนไป การจัดวางระนาบในระบบจะเปลี่ยนไปรูปที่ 4.9 แสดงตัวอย่างผลของการเปลี่ยนของเวคเตอร์ตั้งฉากของระนาบภาพ

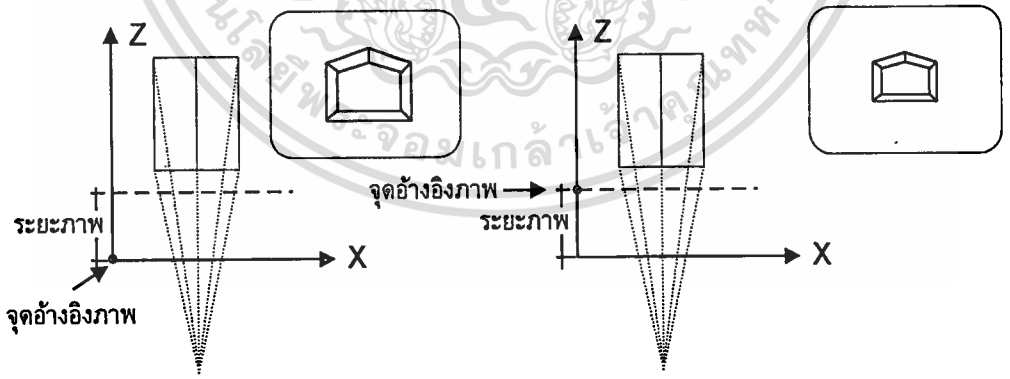
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.9 แสดงการเปลี่ยนเวคเตอร์ตั้งฉากของระนาบภาพ



ถ้าเปลี่ยนระยะภาพ จะทำให้ได้ภาพที่มีขนาดใหญ่เล็กต่างกัน ดังแสดงในรูปที่ 4.10 จะสังเกตได้ว่ารูปที่ 4.10 นั้นเป็นภาพจากการทำโปรเจกชั่นแบบเปอร์สเป็คทีฟถ้าเป็นการทำโปรเจกชั่นแบบขนานนั้น ขนาดของภาพจะไม่ขึ้นอยู่กับระยะภาพ

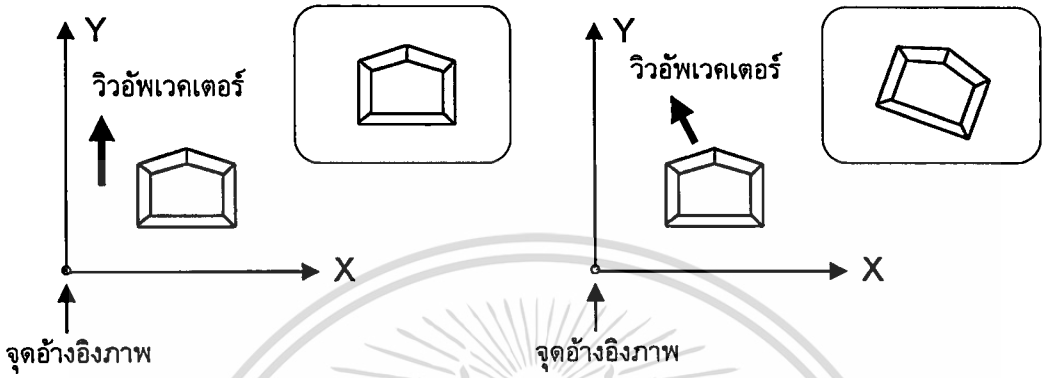
รูปที่ 4.10 แสดงการเปลี่ยนระยะภาพ



การเปลี่ยนทิศทางของวิวอัพเวคเตอร์ เปรียบได้กับการที่เราเอียงกล้องถ่ายภาพไปมานั่นเอง ทำให้ภาพที่ได้มีการเอนเอียงต่างกัน ดังรูปที่ 4.11

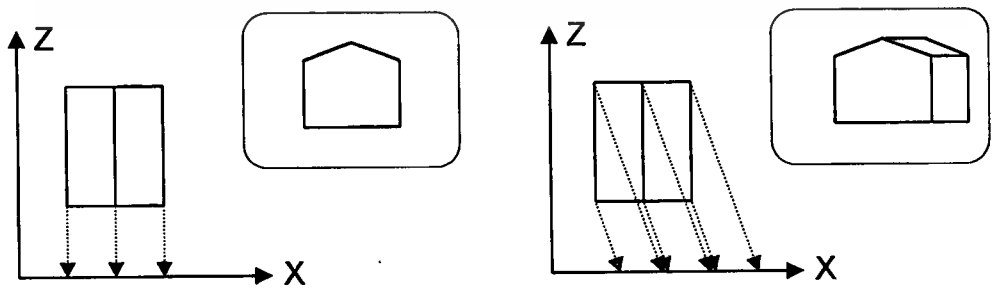
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.11 แสดงการเปลี่ยนวิวอัพเวคเตอร์



พารามิเตอร์ที่กล่าวมาเป็นพารามิเตอร์ที่เกี่ยวข้องกับการมองวัตถุ เสมือนว่าเราจะมองวัตถุในแง่มุมไหน ลักษณะใด นอกจากพารามิเตอร์ทั้ง 4 ที่กล่าวมาแล้วยังมีพารามิเตอร์ที่เกี่ยวกับโปรเจคชั่นของระบบอีกด้วย สำหรับโปรเจคชั่นแบบขนาน ต้องกำหนดทิศทางของแสงที่ฉาย หรือ ทิศทางโปรเจคชั่น (Projection Direction) Dp=[Xp Yp Zp] ผลของการเปลี่ยนทิศทางโปรเจคชั่น แสดงไว้ในรูปที่ 4.12

รูปที่ 4.12 แสดงการเปลี่ยนทิศทางโปรเจคชั่น



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างการทำโปรเจคชั่น

โดยทั่วไปการทำโปรเจคชั่นแบบขนาน มักจะใช้ระนาบ XY เป็นระนาบภาพ และ ทิศทางโปรเจคชั่นจะมีทิศทางขนานกับแนวแกน Z ดังนั้นเงาของจุดต่างๆ ที่เกิดขึ้นบนระนาบภาพจะมีค่าโคออร์ดิเนท X และ Y ที่ไม่เปลี่ยนแปลง เพราะทิศทางโปรเจคชั่นลากตรงจากจุดต่างๆขนานกับแกน Z ลงมาตัดกับระนาบ XY ทำให้การทำโปรเจคชั่นนั้นง่ายมาก คือ เพียงแต่ตัดค่าโคออร์ดิเนท Z ของจุดเหล่านั้นออก เราก็จะได้ตำแหน่งเงาของจุดนั้นๆ บนระนาบภาพ เช่น จุด P(X,Y,Z) จะทำให้เกิดจุด P'(X,Y) บนระนาบภาพ

ในบางครั้งถ้าทิศทางของโปรเจคชั่นเปลี่ยนไปในทิศทางอื่น วิธีดังกล่าวก็ใช้งานไม่ได้ ถ้าทิศทางของโปรเจคชั่นเป็นไปตามทิศทางของเวคเตอร์ [Xp,Yp,Zp] แต่ยังคงใช้ระนาบ XY เป็นระนาบภาพ จะได้ว่าจุด P(X,Y,Z) จะมีเงาหรือ ทำให้เกิดจุด P'(X',Y') บนระนาบภาพ ซึ่ง

$$X' = X - Z(\frac{Xp}{Zp})$$

$$Y' = Y - Z(\frac{Yp}{Zp})$$

หรือเขียนในรูปเมตริกได้ คือ

$$[x' \quad y' \quad z' \quad 1] = [x \quad y \quad z \quad 1] \begin{vmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ \frac{x_p}{z_p} & \frac{y_p}{z_p} & 0 & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix}$$

โดยที่ Z' คือ โคออร์ดิเนท Z ของจุด P' ซึ่งเราตัดทิ้ง และ Z' นี้จะมีค่าเป็นศูนย์เสมอ เพราะเงาโปรเจคชั่นไปบนระนาบ XY (ระนาบ Z=0)

4.1.3 โปรแกรมส่วนติดต่อกับผู้ใช้งาน

สำหรับส่วนที่ติดต่อกับผู้ใช้โปรแกรมนั้น ในการออกแบบโปรแกรมในแต่ละส่วน มีการออกแบบโดยแบ่งออกเป็นโหมดการทำงานต่างๆ ดังนี้

1 โหมดการเขียนและแก้ไขโปรแกรมควบคุมเครื่องจักร

2 โหมดทำงานอัตโนมัติ

3 โหมดความจำ

4 โหมดควบคุมด้วยตัวเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยแต่ละโหมดดังกล่าวข้างต้นมีรายละเอียดแต่ละฟังก์ชั่นดังนี้

1 โหมดการเขียนและแก้ไขโปรแกรม

ส่วนนี้เป็นส่วนของการสร้างโปรแกรมควบคุมเครื่องจักรแบบอัตโนมัติ จากไฟล์รูปภาพจากโปรแกรมวาดแบบ สร้างเป็น G-Code ของทางเดินหัวกัดให้ได้ตามภาพจากไฟล์รูปภาพ ส่วนการแก้ไขก็เป็นการกำหนดค่าความเร็วในการกัดชิ้นงานให้เหมาะสมกับวัตถุ

2 โหมดอัตโนมัติ (Automatic mode)

ในโหมดดังกล่าวเป็นส่วนของการอ่านข้อมูล การแปลงข้อมูล และการทำงานตามข้อมูลที่ได้รับ โดยโปรแกรมจะทำการอ่านข้อมูลที่มาในรูปไฟล์ตัวอักษรทีละบรรทัด แล้วแปลความหมายแล้วปฏิบัติตามคำสั่งทีละบรรทัด และจะวนรอบการทำงานไปจนกว่าจะจบโปรแกรม ซึ่งในโหมดนี้จะมีความสามารถในการทำงานดังนี้

-สามารถกระทำการจำลองการทำงานของเครื่องจักรก่อนทำงานจริงได้

-สั่งให้โปรแกรมทำงาน (Cycle start) โปรแกรมจะทำงานทั้งหมด และติดต่อกับฮาร์ดแวร์ เพื่อควบคุมการทำงานให้สอดคล้องกับโปรแกรม

-สั่งให้โปรแกรมหยุดทำงาน (cycle stop) เพื่อการหยุดโปรแกรม และ ส่งสัญญาณไปบอกระบบฮาร์ดแวร์ ในการควบคุมมอเตอร์หยุดการทำงาน ถ้าจะเริ่มการทำงานใหม่ก็จะต้องโหลดโปรแกรมเข้าไปใหม่

-สั่งให้โปรแกรมทำงานทีละช่วง (Single block) โดยการกำหนดบรรทัดเริ่มต้น และสิ้นสุดให้แก่โปรแกรม ซึ่งโปรแกรมส่วนนี้จะมีประโยชน์ในกรณีที่เราต้องการทำชิ้นงานเฉพาะแค่บางส่วนเท่านั้น ทำให้ไม่ต้องรอให้โปรแกรมทำงานทั้งหมดได้

3 โหมดความจำ (Memory mode)

ในโหมดนี้เป็นส่วนของการปรับเปลี่ยนค่าพารามิเตอร์ของเครื่อง ทั้งในระหว่างการทำงาน และเวลาบังคับเอง โดยในโหมดนี้จะมีความสามารถต่างๆ ดังนี้

-ปรับเปลี่ยนค่า Feed rate คือ การเปลี่ยนค่าความเร็วในการเดินของหัวกัดในการกัดชิ้นงานได้ เฉพาะในกรณีที่โปรแกรมที่นำมาทำการกัดนั้นไม่มีการตั้งค่า Feed rate ไว้

-ปรับเปลี่ยนค่า Spindle rate คือ การเปลี่ยนค่าความเร็วในการหมุนของหัวกัดได้ ใช้ได้ทั้งในโหมดการบังคับเอง และในโหมดอัตโนมัติในกรณีที่ในโปรแกรมไม่ได้ตั้งค่าความเร็วในการหมุนของหัวกัดไว้

-ปรับเปลี่ยนค่า Jog speed คือ การเปลี่ยนค่าความเร็วในการเดินของหัวกัดในโหมดการบังคับเองโดยผู้ใช้โปรแกรม

4 โหมดแบบบังคับเอง (Manual mode)

ในโหมดนี้จะเป็นการบังคับการเคลื่อนที่ของเครื่องจักรในแบบธรรมดา โดยอาศัยการกดปุ่ม โดยจะสามารถสั่งการเคลื่อนที่ได้ทีละแกน ในโหมดนี้จะมีความสามารถต่างๆ ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

-ควบคุมการเคลื่อนที่ของแกนต่างๆ โดยจะสามารถควบคุมได้ทีละ 1 แกนเท่านั้น ด้วยความเร็วที่ได้กำหนดไว้ในโหมดความจำ

-กลับสู่ตำแหน่งเริ่มต้น (Home) โดยจะสั่งให้มอเตอร์ทั้ง 3 แกนเคลื่อนที่กลับไปยังตำแหน่ง (0,0,0) ของเครื่องทันที

-เซ็ตศูนย์โปรแกรม (Set Zero) โดยจะทำการเซ็ตค่า ณ ตำแหน่งนั้นๆ ให้เป็นตำแหน่งศูนย์ของโปรแกรม

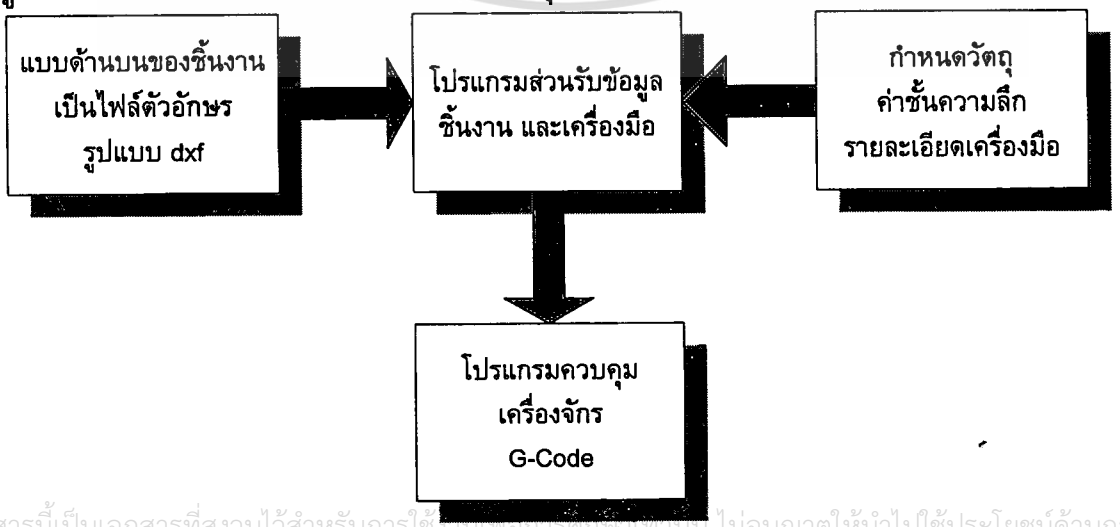
-สั่งให้หัวหมุน ทำการหมุนด้วยความเร็วที่กำหนดไว้ และสั่งให้หยุดหมุนได้

-หยุดฉุกเฉิน (Emergency Stop) เป็นการหยุดการทำงานของเครื่องในกรณีฉุกเฉิน โดยได้มีการทำฮาร์ดแวร์อยู่ภายนอก โดยให้ทุกอย่างหยุดทำงาน

4.2 โปรแกรมส่วนของการรับข้อมูลชิ้นงานและเครื่องมือ

ส่วนนี้จะเป็นการวาดรูปชิ้นงานจากโปรแกรม โดยทำการกำหนดขนาดทั้งความกว้าง ความยาว และความสูงของชิ้นงานที่จะทำการกัด พร้อมทั้งทำการกำหนดลักษณะ และขนาดของเครื่องมือที่ใช้ในการกัดขึ้นมา จากนั้นจะทำการโหลดแบบงานที่วาดจากโปรแกรมวาดแบบเข้ามาในโปรแกรม (แบบงานเป็นไฟล์ตัวอักษร นามสกุล DXF) โดยแบบที่วาดจะเป็นแบบของมุมมองด้านบนของชิ้นงาน เป็นส่วนที่จะให้เครื่องกัดทำการกัดให้เป็นรูปที่ต้องการ จากนั้นโปรแกรมจะทำการวาดรูปผิวด้านบนของชิ้นงาน เพื่อให้ผู้ใช้งานทำการกำหนดความลึกของบริเวณที่จะทำการกัด เมื่อผู้ใช้กำหนดความลึกเสร็จแล้วเครื่องก็จะทำการคำนวณหาแนวกัดตามค่าที่กำหนดไว้ เพื่อสร้างโปรแกรมในส่วนที่ใช้ควบคุมเครื่องจักรขึ้นมา และทำการบันทึกโปรแกรมในส่วนควบคุมเครื่องจักรที่สร้างขึ้นมานั้น ลงในแผ่นบันทึกของเครื่องคอมพิวเตอร์สำหรับไว้ใช้งานต่อไป

รูปที่ 4.13 แสดงโปรแกรมส่วนสร้างโค้ดควบคุมเครื่องจักร



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

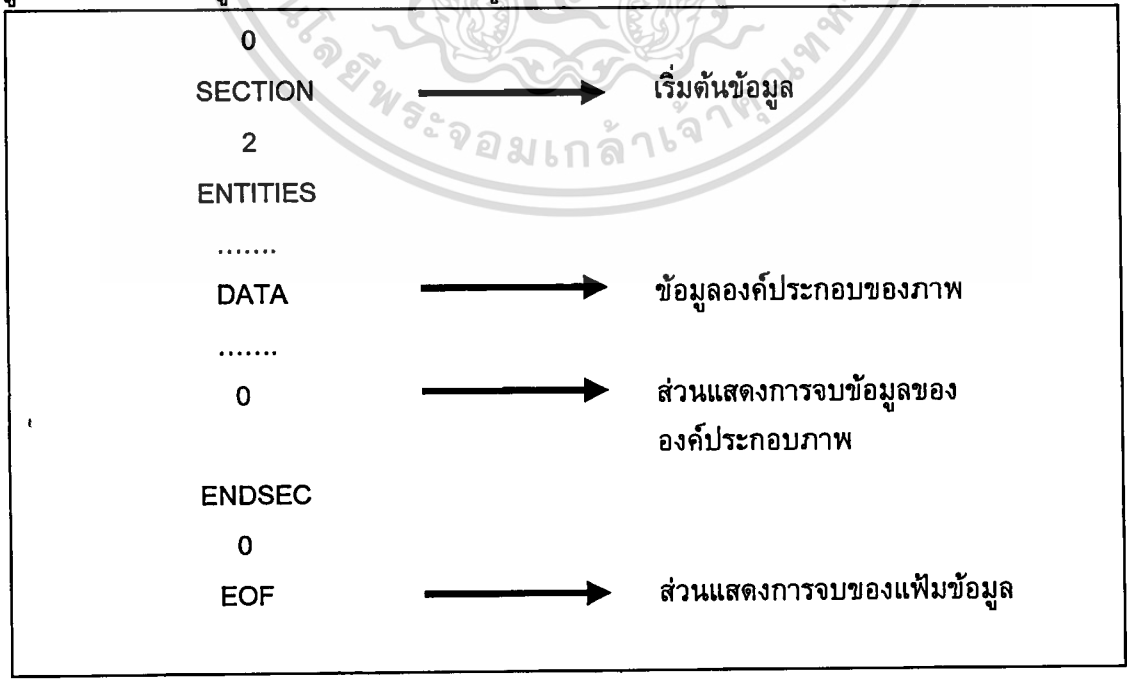
4.2.1 การวาดแบบของชิ้นงาน

ในส่วนของการวาดแบบชิ้นงานจะใช้โปรแกรมวาดแบบที่มีอยู่สร้างในส่วนแบบของชิ้นงานเฉพาะด้านที่มองจากด้านบนลงมา แล้วทำการบันทึกแบบที่วาดขึ้นเก็บไว้ในแผ่นบันทึกในรูปแบบของไฟล์ตัวอักษรที่มีนามสกุล dxf เพื่อที่จะได้นำไฟล์แบบด้านบนของชิ้นงานไปใช้งานต่อไป

4.2.2 การรับข้อมูลไฟล์จากโปรแกรมจำพวก CAD

โปรแกรมวาดแบบที่เป็นมาตราฐานทั่วไป จะเก็บข้อมูลจากภาพที่วาดขึ้น ในรูปแบบที่สามารถนำไปเชื่อมต่อกับโปรแกรมวาดแบบมาตรฐานอื่นๆได้ ซึ่งข้อมูลทั้งหมดจะอยู่ในรูปของแฟ้มข้อมูลเอกสาร (Text File) ที่เป็นมาตรฐานเรียกว่า ASCII Drawing Interchange File (DXF) แบ่งออกได้เป็น 5 ส่วน คือ ส่วนข้อมูลทั่วไป (HEADER section) ส่วนตารางรายชื่อ (TABLE section) ส่วนข้อมูลภาพชุด (BLOCKS section) ส่วนข้อมูลที่เป็นองค์ประกอบของภาพ (ENTITIES section) และส่วนการจบไฟล์ (EOF section) สำหรับงานวิจัยนี้พบว่าในการแปลงข้อมูลภาพ ให้เป็นโปรแกรมสั่งงานเครื่องซีเอ็นซีนั้น จะใช้เพียงส่วนข้อมูลที่เป็นองค์ประกอบภาพเท่านั้น เพราะข้อมูลส่วนนี้เป็นโครงสร้างแต่ละส่วนของภาพทั้งหมด ซึ่งเป็นการเพียงพอสำหรับนำไปสร้างเป็นโปรแกรมสั่งงานเพื่อควบคุมเครื่องซีเอ็นซี ให้ทำงานตามที่ออกแบบไว้ ดังนั้นจะพิจารณาเฉพาะรูปแบบข้อมูลในส่วนองค์ประกอบภาพเท่านั้น ลักษณะของการเก็บบันทึกข้อมูลในส่วนนี้จะมีรูปแบบมาตรฐาน ดังรูปที่ 4.14

รูปที่ 4.14 แสดงรูปแบบการเก็บส่วนข้อมูลที่เป็นองค์ประกอบของภาพในไฟล์ DXF



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในบรรทัดที่ 1 จะใช้เลข "0" เป็นตัวกำหนดจุดเริ่มต้นของส่วนข้อมูลทั้งหมด ได้แก่ "SECTION" ในบรรทัดต่อมา ส่วนต่อมาจะใช้เลข "2" เป็นตัวกำหนดจุดเริ่มต้นของส่วนองค์ประกอบของภาพได้แก่ "ENTITIES" ในบรรทัดต่อมา ข้อมูล "DATA" ดังกล่าว จะประกอบด้วย เส้นตรง (LINE) เส้นโค้ง (ARC) วงกลม (CIRCLE) และ เส้นต่อเนื่อง (POLYLIINE) ปิดท้ายชุดข้อมูลด้วยตัวเลข "0" ตามด้วยคำสั่ง "SECTION" เพื่อแสดงถึงการจบข้อมูลในส่วนขององค์ประกอบภาพ และทำการปิดไฟล์ด้วยตัวเลข "0" ตามด้วยคำสั่ง "EOF" ในบรรทัดสุดท้าย จะเห็นว่ารูปแบบของแฟ้มข้อมูลในไฟล์ DXF จะใช้รหัสตัวเลข (group code) คู่กับคำสั่ง หรือค่าตัวแปร (group value) รหัสตัวเลขจะถูกกำหนดคงที่เพื่อแสดงถึงชนิด และความหมายของคำสั่งหรือค่าตัวแปรที่ตามมา โดยตัวแปรจะอยู่ในรูปของเลขทศนิยม และคำสั่งจะอยู่ในรูปของกลุ่มตัวอักษร หรือ เลขจำนวนเต็ม ต่อไปจะเป็นเรื่องของรายละเอียดในส่วนข้อมูลเส้นตรง เส้นโค้ง วงกลม และเส้นต่อเนื่อง ซึ่งเป็นส่วนสำคัญในการแปลงเป็นชุดคำสั่งควบคุมเครื่องซีเอ็นซี

4.2.2.1 เส้นตรง

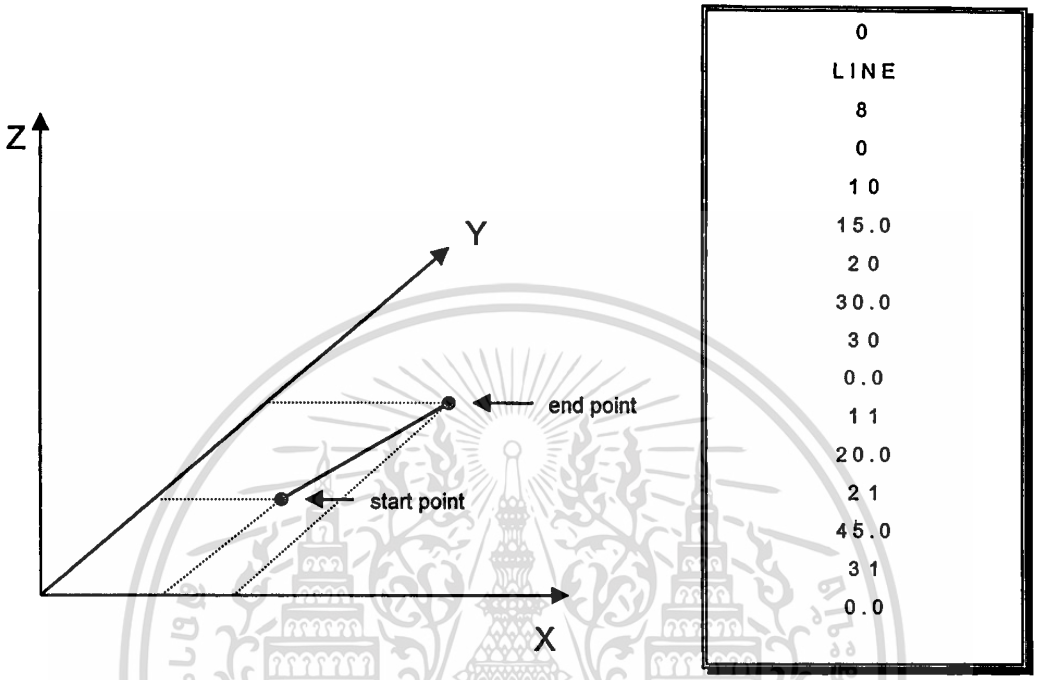
ข้อมูลเส้นตรงจะเริ่มต้นด้วยรหัสเลข "0" คู่กับคำสั่ง "LINE" ตามด้วยข้อมูลของเส้น ดังรูปที่ 4.15 จุดเริ่มต้นของเส้นตรงกำหนดด้วยรหัสเลข "10" "20" และ "30" ที่แสดงพิกัดของจุดเริ่มต้นในทิศทางของแกน X แกน Y และแกน Z ตามลำดับ ส่วนจุดสิ้นสุดของเส้นตรงกำหนดด้วยรหัสตัวเลข "11" "21" และ "31" แสดงพิกัดของจุดสิ้นสุดในทิศทางของแกน X แกน Y และแกน Z ตามลำดับ ดังตัวอย่างรูปที่ 4.16

รูปที่ 4.15 แสดงแบบการเก็บข้อมูลเส้นตรงของไฟล์ DXF

```
0
LINE
ชั้นเก็บข้อมูล
10
ค่าจุดเริ่มต้น แกน X
20
ค่าจุดเริ่มต้น แกน Y
30
ค่าจุดเริ่มต้น แกน Z
11
ค่าจุดปลาย แกน X
21
ค่าจุดปลาย แกน Y
31
ค่าจุดปลาย แกน Z
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.16 แสดงตัวอย่างข้อมูลเส้นตรงในรูปแบบ dxf



4.2.2.2 เส้นโค้ง

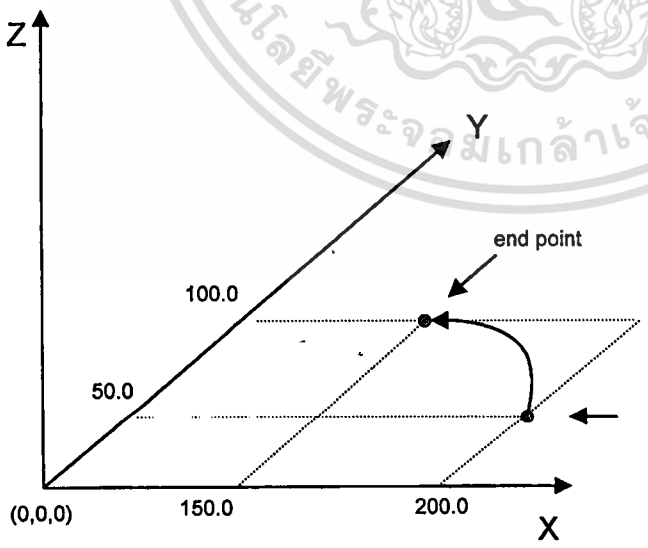
ข้อมูลเส้นโค้งจะเริ่มต้นด้วยรหัสเลข "0" คู่กับคำสั่ง "ARC" ตามด้วยข้อมูลของเส้น ดังรูปที่ 4.17 โดยใช้รหัสเลข "10" "20" และ "30" แสดงพิกัดของจุดศูนย์กลางของเส้นโค้งบนแกน X แกน Y และแกน Z ตามลำดับ ใช้รหัส "40" บอกขนาดของรัศมีของเส้นโค้ง ใช้เลขรหัส "50" บอกขนาดมุมเริ่มต้น และใช้รหัส "51" บอกขนาดของมุมสิ้นสุด โดยขนาดของมุมทั้งสองจะวัดจากแกน X กวาดไปในทิศทวนเข็มนาฬิกามีหน่วยเป็นองศา ตัวอย่างเก็บข้อมูลเส้นโค้งแสดงในรูปที่ 4.18

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.17 แสดงแบบการเก็บข้อมูลเส้นโค้งของไฟล์ DXF

```
0
ARC
8
ชั้นเก็บข้อมูล
10
จุดศูนย์กลางเทียบกับแกน X
20
จุดศูนย์กลางเทียบกับแกน Y
30
จุดศูนย์กลางเทียบกับแกน Z
40
รัศมีของเส้นโค้ง
50
มุมเริ่มต้นของเส้นโค้งเทียบแกน X
51
มุมสิ้นสุดของเส้นโค้งเทียบแกน X
```

รูปที่ 4.18 แสดงตัวอย่างของส่วนโค้งในรูปแบบ dxf




```
0
ARC
8
0
10
150.0
20
50.0
30
0.0
40
50.0
50
0.0
51
90.0
```

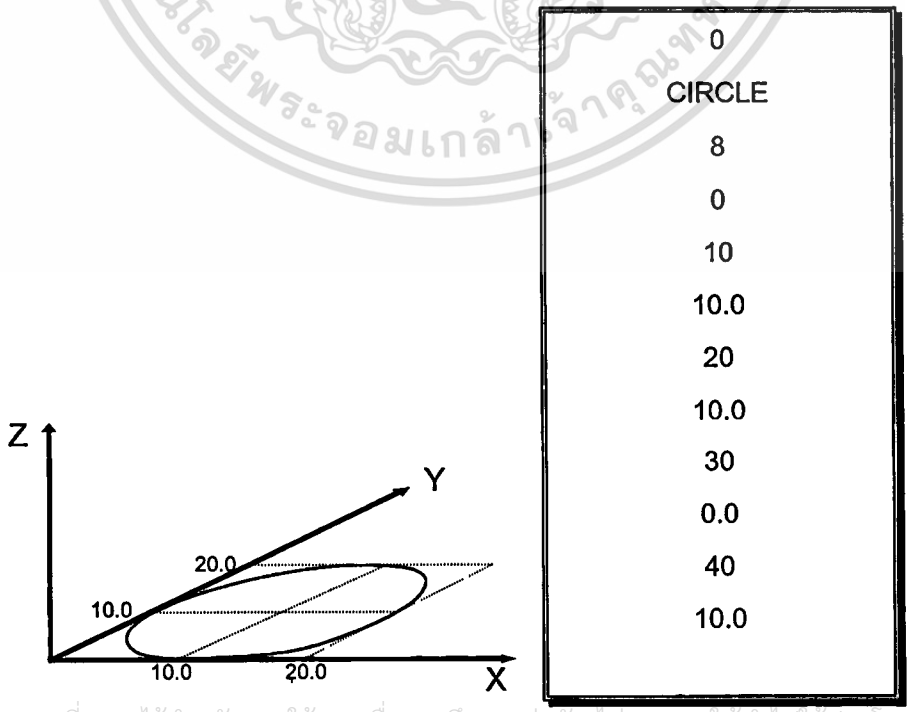
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.2.3 วงกลม

ข้อมูลเส้นโค้งจะเริ่มต้นด้วยรหัสเลข "0" คู่กับคำสั่ง "CIRCLE" ตามด้วยข้อมูลของเส้น ดังรูปที่ 4.19 โดยใช้รหัสเลข "10" "20" และ "30" แสดงพิกัดของจุดศูนย์กลางของวงกลมบนแกน X แกน Y และแกน Z ตามลำดับ ใช้รหัส "40" บอกขนาดของรัศมีของวงกลม ตัวอย่างเก็บข้อมูลเส้นโค้งแสดงในรูปที่ 4.20

รูปที่ 4.19 แสดงรูปแบบการเก็บข้อมูลวงกลมของไฟล์ DXF

รูปที่ 4.20 แสดงตัวอย่างของวงกลมในรูปแบบของ dxf

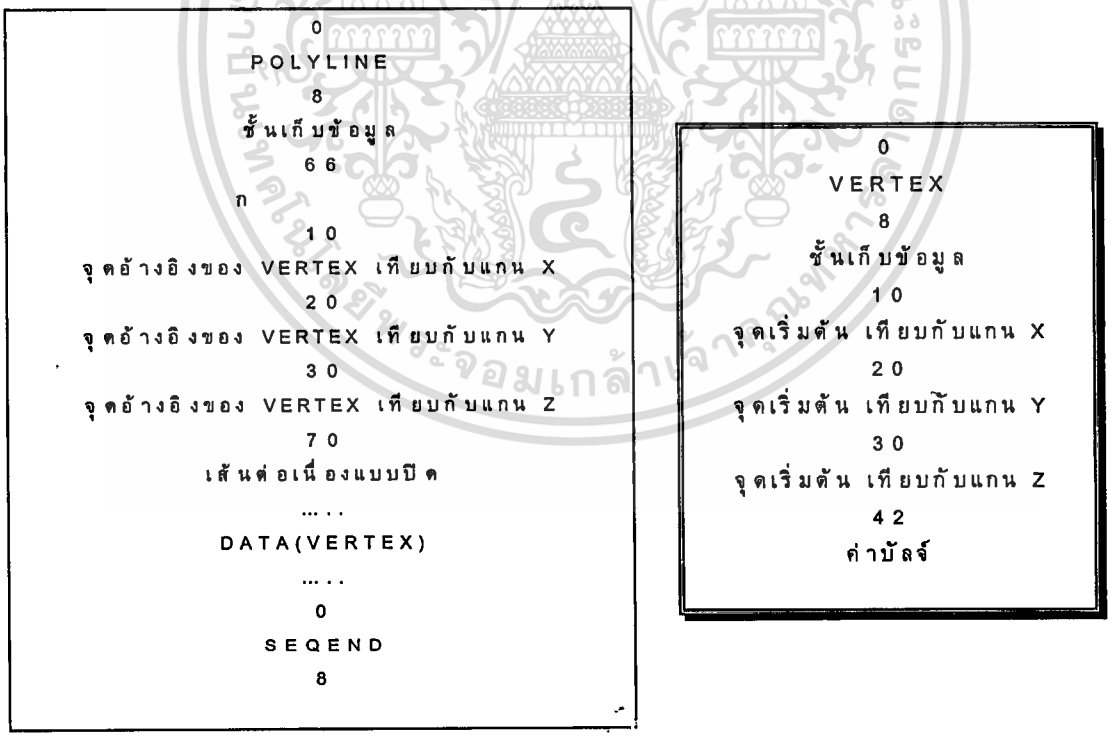


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.2.4 เส้นต่อเนื่อง

เส้นต่อเนื่องในโปรแกรม หมายถึง เส้นองค์ประกอบภาพที่ผู้วาดเพียงแต่ทำการกำหนดจุดตัด หรือ จุดเชื่อมต่อของเส้นแต่ละเส้น แล้วทำการลากเส้นเชื่อมจุดต่อเหล่านั้นให้เป็นรูปภาพของชิ้นงาน โดยเส้นในแต่ละส่วนอาจเป็นเส้นตรง หรือ เส้นโค้ง ดังนั้นรูปแบบการเก็บบันทึกข้อมูลของเส้นตรง หรือ เส้นโค้งที่ได้จากเส้นต่อเนื่องจึงแตกต่างไปจากเดิม ข้อมูลเส้นต่อเนื่องจะเริ่มต้นด้วยรหัสเลข "0" คู่กับคำสั่ง "POLYLINE" ตามด้วยข้อมูลเส้นต่อเนื่อง ที่มีการแบ่งเป็นส่วนๆ ตามจุดเชื่อมต่อ (VERTEX) ของเส้นองค์ประกอบภาพแต่ละเส้นโดยเริ่มต้นข้อมูลของแต่ละส่วนด้วยคำสั่ง "VERTEX" ดังแสดงในรูปที่ 4.21 ข้อมูลภาพใต้คำสั่ง "VERTEX" จะเป็นข้อมูลเส้นต่อเนื่องในส่วนของเส้นตรงหรือเส้นโค้ง โดยใช้รหัสเลข "10 "20 และ "30" บอกจุดเริ่มต้นของเส้นทั้งสองเมื่อเทียบกับแกน X แกน Y และ Z ตามลำดับ และใช้รหัสตัวเลข "42" บอกค่าความนูน (Bulge) ของส่วนเส้นตรง หรือ ส่วนเส้นโค้ง ดังแสดงในรูปที่ 4.22

รูปที่ 4.21 แสดงข้อมูลเส้นต่อเนื่อง และข้อมูลในส่วนจุดเชื่อมต่อ (VERTEX) ของแต่ละเส้น



สำหรับส่วนของเส้นตรงจะมีค่าความนูนเป็นศูนย์เสมอ ส่วนเส้นโค้งจะมีค่าความนูนที่ติดระหว่างจุดปลายทั้งสองของส่วนโค้งดังนี้ เมื่อลากเส้นตรงเชื่อมระหว่างจุดปลายทั้งสองของ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

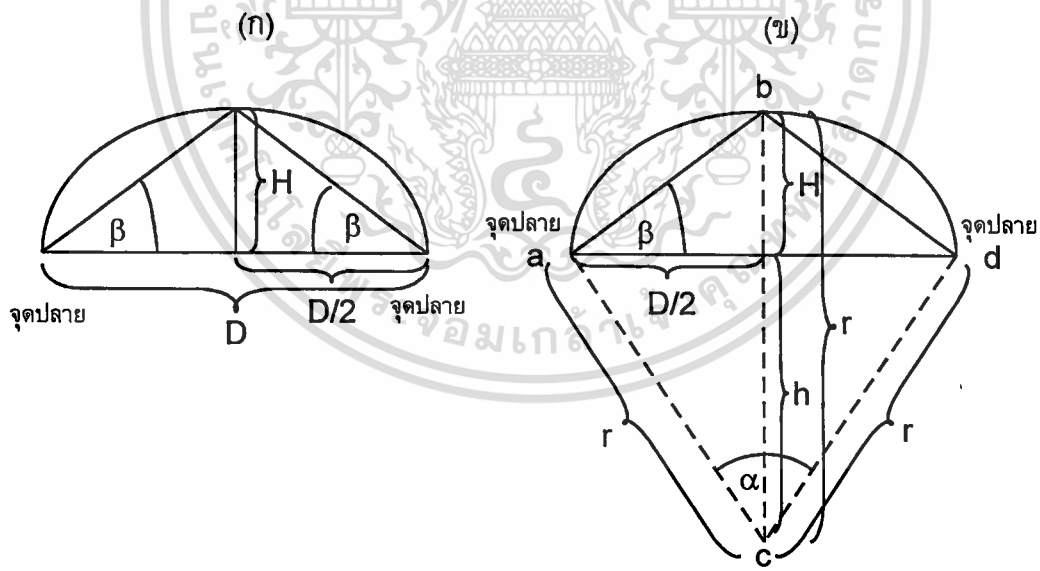
ส่วนโค้ง สมมุติให้มีความยาวเป็น D และลากเส้นตั้งฉากจากจุดกึ่งกลางของเส้นตรงนี้ไปตัดกับส่วนโค้ง สมมุติให้เส้นตั้งฉากนี้มีความยาว H ดังแสดงในรูปที่ 4.22 (ก) ถ้าลากเส้นตรงจากจุดปลายทั้งสองไปยังจุดตัดระหว่างส่วนโค้งกับเส้นตั้งฉากจะทำให้เกิดมุม β ดังนั้น ถ้าให้ B เป็นค่าความนูนของส่วนโค้งตามนิยามของโปรแกรมมาตรฐานจะได้

$$B = \tan(\beta)$$

นั่นคือ

$$B = \frac{H}{(D/2)} = 2H/D$$

รูปที่ 4.22 แสดงการคำนวณหาค่าความนูน (BULGE) ของส่วนโค้ง



ถ้าลากเส้นตั้งฉากดังกล่าวตัดจุดกึ่งกลางของเส้นตรงที่เชื่อมจุดปลายทั้งสอง ให้มีความยาวทั้งหมดเท่ากับรัศมีของเส้นโค้ง และเชื่อมปลายเส้นตั้งฉากนี้กับจุดปลายทั้งสองของส่วนโค้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งเส้นทั้งสองจะมีความยาวเท่ากับรัศมีของส่วนโค้งเช่นกัน ดังรูป 4.22 (ข) ถ้ากำหนดให้ r เป็นรัศมีของส่วนโค้ง จุด c เป็นจุดศูนย์กลางของส่วนโค้ง เนื่องจากเส้นตรง bc เป็นเส้นตั้งฉากกับจุดกึ่งกลางของเส้นตรง ad จึงเป็นเส้นแบ่งครึ่งมุม acd ดังนั้น ถ้ากำหนดให้

$$\alpha = \angle acd$$

จะได้

$$\angle acd = \angle dcb = \frac{\alpha}{2}$$

และจากรูป 4.22 (ข) กำหนดให้

$$H = r - h$$

จากสมการ (2) และ (4) จะได้

$$B = \frac{r-h}{D/2}$$

$$= \frac{r}{D/2} - \frac{h}{D/2}$$

$$= \frac{1}{\sin(\alpha/2)} - \frac{1}{\tan(\alpha/2)}$$

$$= \frac{1}{\sin(\alpha/2)} - \frac{\cos(\alpha/2)}{\sin(\alpha/2)}$$

$$= \frac{1-\cos(\alpha/2)}{\sin(\alpha/2)}$$

เพราะฉะนั้นจะได้

$$B = \tan(\alpha/4)$$

ในที่นี้ α จะเป็นมุมที่รองรับส่วนโค้งที่เกิดจากการลากเส้นตรงที่มีความยาวเท่ารัศมีของส่วนโค้งจากปลายทั้งสองของเส้นโค้งให้ไปจดกัน ณ.จุดศูนย์กลางของส่วนโค้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

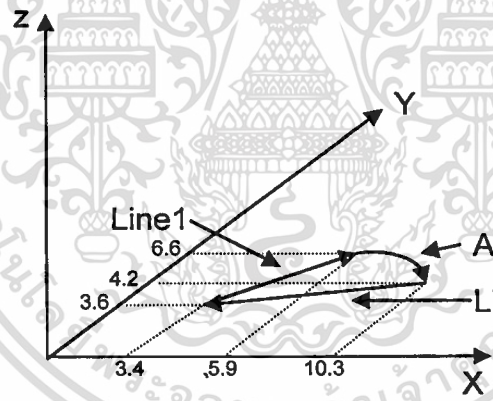
ในโปรแกรมวาดแบบมาตรฐานทั่วไปจะทำการคำนวณหาค่า B จากสมการ 3.2 หรือ สมการ 3.5 และเมื่อได้ค่า B แล้วจะมีการกำหนดเครื่องหมาย + และ - ให้กับค่า B เพื่อบอกทิศทางของการลากเส้นโค้งจากจุดเริ่มต้นไปจุดปลายทาง เมื่อเทียบกับจุดศูนย์กลางของเส้นโค้งดังนี้

ก) ถ้าลากเส้นโค้งแบบทวนเข็มนาฬิกาจะมีเครื่องหมาย +

ข) ถ้าลากเส้นโค้งแบบตามเข็มนาฬิกาจะมีเครื่องหมาย -

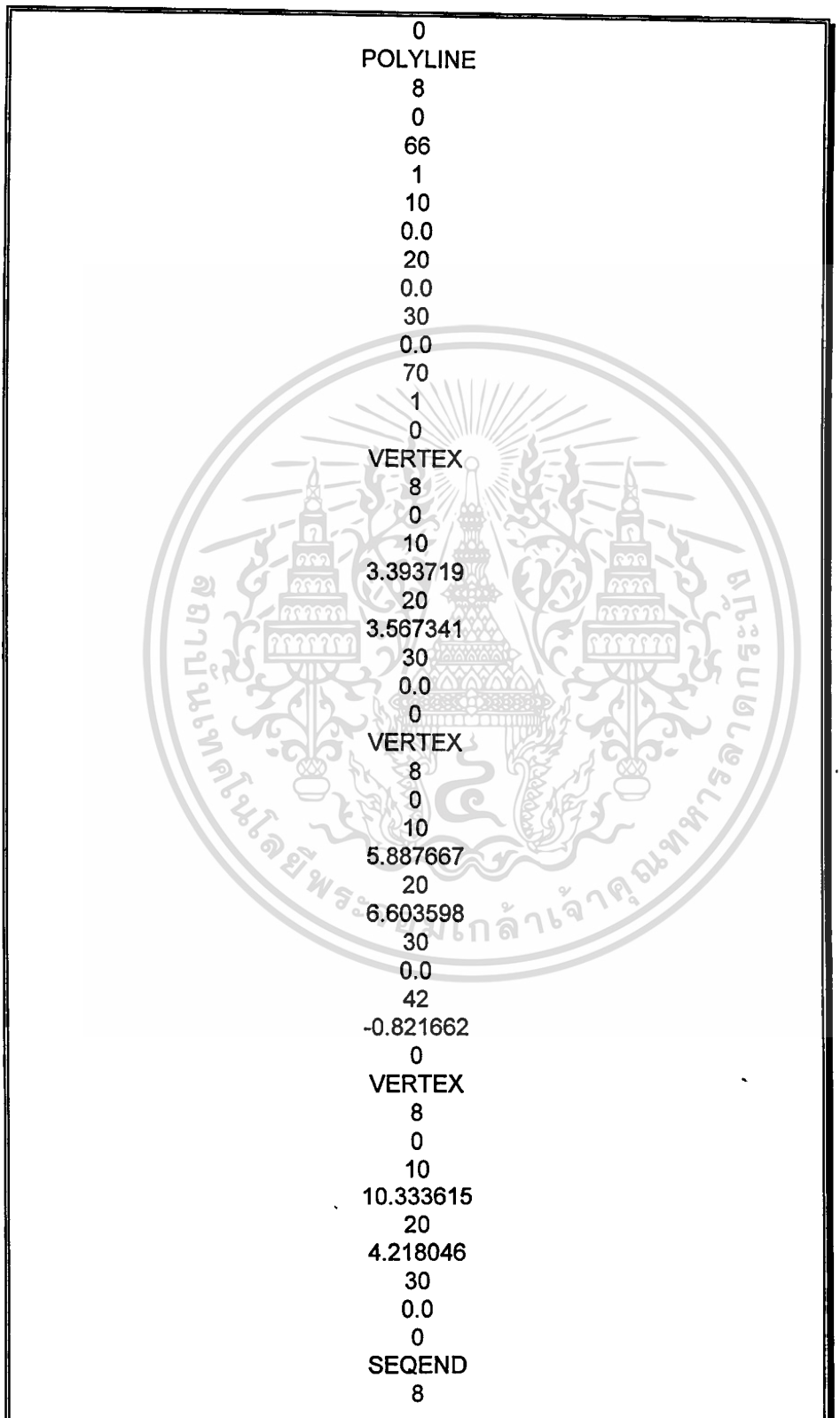
ในการสร้างภาพชิ้นงานด้วยคำสั่งเส้นต่อเนื่องในโปรแกรมวาดแบบ เส้นแต่ละเส้นจะถูกสร้างขึ้นในลักษณะต่อเนื่องกันเสมอ ดังนี้ข้อมูลของแต่ละเส้นขององค์ประกอบภาพจะถูกเก็บบันทึกเรียงตามลำดับการถูกสร้างก่อนหลังในโปรแกรมวาดแบบ ดังแสดงในตัวอย่างรูปที่ 4.23 และ รูปที่ 4.24

รูปที่ 4.23 แสดงเส้นต่อเนื่อง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.24 แสดงตัวอย่างข้อมูลภาพต่อเนื่อง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 การแปลงข้อมูลภาพชิ้นงานเป็นโปรแกรมสั่งงานเครื่องซีเอ็นซี

จากส่วนรายละเอียดภาพดังที่กล่าวมา พบว่าจะต้องดัดแปลงข้อมูลบางส่วนก่อนเพื่อความเหมาะสม แล้วทำการสร้างเป็นชุดคำสั่งเริ่มต้นเพื่อนำไปแปลงเป็นโปรแกรมสั่งงานเครื่องซีเอ็นซีต่อไป โดยเฉพาะข้อมูลในส่วนของเส้นต่อเนื่องนั้น จะต้องทำการแยกข้อมูลเป็นส่วนของเส้นตรง และเส้นโค้ง เมื่อได้ข้อมูลองค์ประกอบภาพที่จำเป็นแล้วก็แปลงให้อยู่ในรูปของชุดคำสั่งของการสร้างภาพชิ้นงาน ที่มีการเรียงลำดับไปทีละเส้นในทิศทางต่อเนื่องกัน

ในโครงงานนี้ได้กำหนดชุดคำสั่งเริ่มต้น 3 แบบ ได้แก่คำสั่งเส้นตรง คำสั่งเส้นโค้ง และคำสั่งวงกลม

คำสั่งเส้นตรง Line(Xs,Ys,Xe,Ye)

คำสั่งเส้นโค้ง Arc(Xc,Yc,Xs,Ys,Xe,Ye,d)

คำสั่งวงกลม Circle(Xc,Yc,r)

โดยที่ Xs : จุดเริ่มต้นของเส้นตรง หรือ เส้นโค้ง เทียบกับแกน X

Ys : จุดเริ่มต้นของเส้นตรง หรือ เส้นโค้ง เทียบกับแกน Y

Xe : จุดสิ้นสุดของเส้นตรง หรือ เส้นโค้ง เทียบกับแกน X

Ye : จุดสิ้นสุดของเส้นตรง หรือ เส้นโค้ง เทียบกับแกน Y

Xc : จุดศูนย์กลางของส่วนโค้ง หรือ ของวงกลมเทียบกับแกน X

Yc : จุดศูนย์กลางของส่วนโค้ง หรือ ของวงกลมเทียบกับแกน Y

r : รัศมีวงกลม

d : ทิศทางของเส้นโค้ง

การกำหนดตัวแปรข้างต้นจะได้จากข้อมูลองค์ประกอบภาพ และจะใช้หลักการทางคณิตศาสตร์ และ เมื่อได้ชุดคำสั่งเริ่มต้นแล้ว จะทำการหาแนวการกัดจากชุดคำสั่งเบื้องต้นดังกล่าวดังนี้

ขั้นตอนการสร้างชุดคำสั่ง

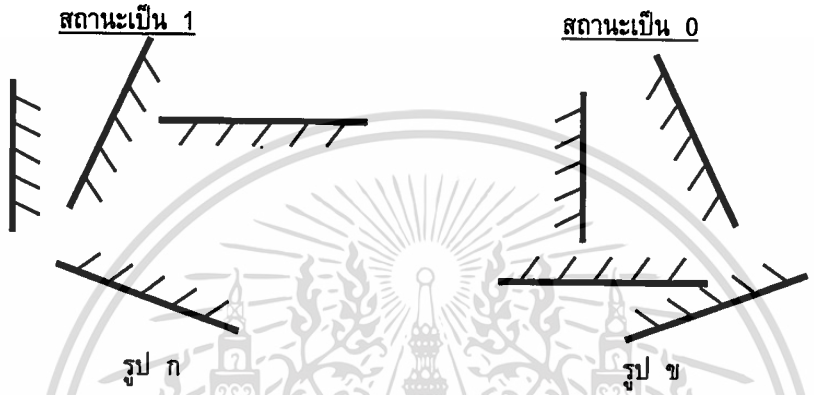
เริ่มจากการกำหนดขนาดของชิ้นงานขึ้นมาก่อน แล้วจึงทำการโหลดแฟ้มข้อมูลภาพชิ้นงานจากโปรแกรมวาดแบบ ซึ่งอยู่ในรูปแบบแฟ้มเอกสารที่มีนามสกุล DXF ซอฟท์แวร์ที่สร้างขึ้นจะทำการดึงข้อมูลจากส่วนขององค์ประกอบภาพขึ้นมา จากนั้นผู้ใช้งานจะต้องกำหนดความลึกของชั้นรูปวงปิดแต่ละรูป แล้วโปรแกรมจะทำการนำข้อมูลที่กำหนดดังกล่าวไปคำนวณหาค่าเส้นทางเดินของเครื่องมือ เพื่อนำไปสร้างเป็นโปรแกรมชุดคำสั่งควบคุมเครื่องกัดซีเอ็นซีอีกต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การกำหนดสถานะของรูปวงปิด

ในการจะหาเส้นทางเดินของเครื่องมือรอบรูปวงปิดใดๆ จะต้องทำการหาสถานะของเส้นรอบรูปวงปิดก่อน โดยกำหนดสถานะของเส้นที่ประกอบกันเป็นรูปวงปิดไว้ 2 สถานะ คือ 1 กับ 0 ดังแสดงในรูปที่ 4.25 โดยเส้นทึบๆ ที่เป็นขีดๆ แทนด้านในของรูปวงปิด

รูปที่ 4.25 แสดงสถานะของเส้นแบบต่างๆ ที่อยู่บนเส้นรอบรูปวงปิด



สถานะของเส้นเป็นดังนี้ คือ

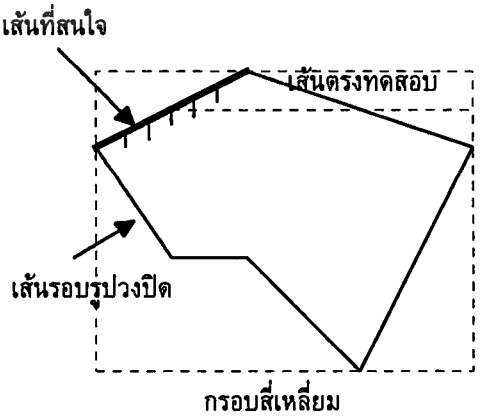
1 เส้นที่มีด้านขวาเป็นด้านในของรูปวงปิดให้มีสถานะเป็น 1 ดังรูป 4.25 ก

2 เส้นที่มีด้านซ้ายเป็นด้านในของรูปวงปิดให้มีสถานะเป็น 0 ดังรูป 4.25 ข

3 เส้นที่อยู่ในแนวนอนที่มีด้านล่างของเส้นเป็นด้านในของรูปวงปิดจะให้มีสถานะเป็น 1 ดังรูป 4.25 ก

4 เส้นที่อยู่ในแนวนอนที่มีด้านล่างของเส้นเป็นด้านในของรูปวงปิดจะให้มีสถานะเป็น 1 ดังรูป 4.25 ข

รูปที่ 4.26 แสดงวิธีการหาสถานะของเส้นที่อยู่บนรูปวงปิด



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วิธีการหาสถานะของเส้นรอบรูปวงปิด

1 กำหนดขอบเขตบริเวณรอบเส้นวงปิด ให้เป็นกรอบสี่เหลี่ยมรอบวงปิด ดังรูป 4.26

2 เลือกเส้นจากรูปวงปิดมาทำการหาสถานะทีละเส้น

3 ถ้าเส้นที่เลือกไม่ได้ขนานกับแกน X ให้ลากเส้นทดสอบออกจากจุดกึ่งกลางเส้นที่ต้องการหาสถานะ ไปยังขอบด้านขวาของเส้นวงปิด

4 ถ้าเส้นที่เลือกขนานกับแกน X ให้ลากเส้นทดสอบออกจากจุดกึ่งกลางเส้นที่ต้องการหาไปยังขอบด้านบนของเส้นวงปิด

5 ให้นับจำนวนเส้นบนรูปวงปิดที่เส้นทดสอบลากผ่านว่ามีกี่เส้น

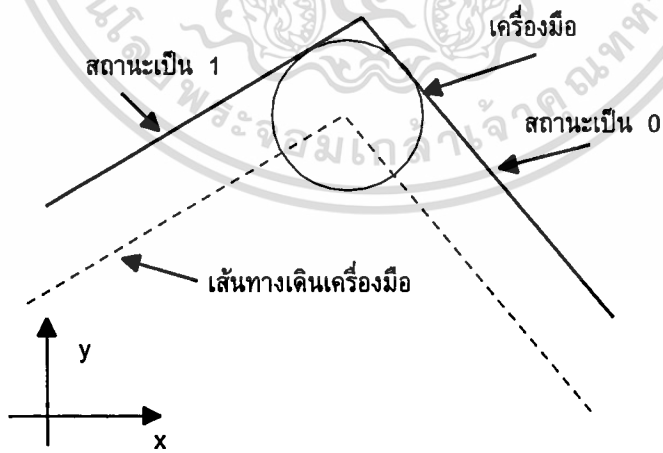
ดังนี้ ถ้าจำนวนเส้นที่เส้นสมมติลากผ่านเป็นเลขคู่ ให้เส้นนั้นมีสถานะเป็น 1

ถ้าจำนวนเส้นที่เส้นสมมติลากผ่านเป็นเลขคี่ ให้เส้นนั้นมีสถานะเป็น 0

การหาเส้นทางเดินของเครื่องมือภายนอก หรือ ภายในเส้นรอบรูปวงปิด

การหาเส้นทางเดินของเครื่องมือจะใช้สถานะของเส้นเป็นการกำหนดว่าให้บวก หรือ ลบค่าออฟเซ็ตของเครื่องมือ ดังนี้

รูปที่ 4.27 แสดงเส้นทางเดินของเครื่องมือภายในเส้นรอบรูปวงปิด



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

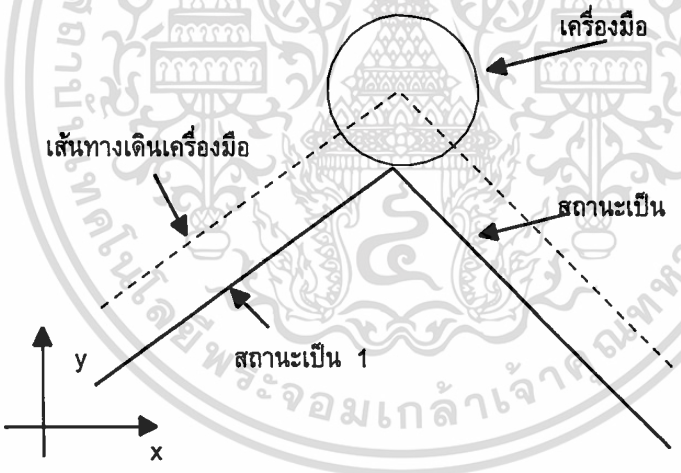
การหาเส้นทางเดินของเครื่องมือในแนวข้างในของเส้นรอบรูปวงปิด

การหาเส้นทางเดินของเครื่องมือจะเป็นการหาเส้นที่ขนานกับเส้นรอบรูปวงปิดที่ต้องการ โดยนำเส้นที่อยู่บนเส้นวงปิดนั้นๆ มาทำการพิจารณาหาเส้นทางเดินทีละเส้น เมื่อเราพิจารณาสถานะของเส้นที่อยู่รอบรูปเส้นวงปิดแล้ว เราจะพบว่าจะมีการเพิ่มหรือลดค่าออฟเซ็ตของเส้นขนานที่สร้างจากเส้นที่นำมาพิจารณาทั้งในแนวแกน x และในแนวแกน Y โดยแบ่งได้เป็น 2 กรณี ดังนี้

1 ในเส้นที่มีสถานะเป็น 1 เส้นทางเดินของเครื่องมือจะเป็นเส้นขนานกับเส้นที่ต้องการหา และจะถูกเพิ่มค่าออฟเซ็ตในแนวแกน X และจะลดค่าออฟเซ็ตในแนวแกน Y

2 ในเส้นที่มีสถานะเป็น 0 เส้นทางเดินของเครื่องมือจะเป็นเส้นขนานกับเส้นที่ต้องการหา และจะถูกลดค่าออฟเซ็ตในแนวแกน x และจะเพิ่มค่าออฟเซ็ตในแนวแกน Y

รูปที่ 4.28 แสดงเส้นทางเดินของเครื่องมือภายนอกเส้นรอบรูปวงปิด



ภารหาเส้นทางเดินของเครื่องมือในแนวภายนอกเส้นรอบรูปวงปิด

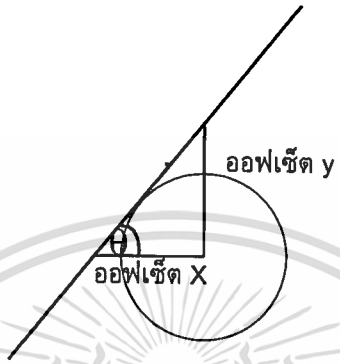
ในการหาเส้นทางเดินของเครื่องมือด้านนอกก็จะพิจารณาในทำนองเดียวกัน กับการหาเส้นทางเดินภายใน แต่คิดแบบตรงกันข้าม โดยใช้การพิจารณาสถานะเส้นที่อยู่บนรูปวงปิด แล้วหาค่าออฟเซ็ตในแนวแกน X และ Y ซึ่งแบ่งได้เป็น 2 กรณี ดังนี้

1 ในเส้นที่มีสถานะเป็น 1 เส้นทางเดินของเครื่องมือจะเป็นเส้นขนานกับเส้นที่ต้องการหา และจะถูกเพิ่มลดออฟเซ็ตในแนวแกน X และจะเพิ่มค่าออฟเซ็ตในแนวแกน Y

2 ในเส้นที่มีสถานะเป็น 0 เส้นทางเดินของเครื่องมือจะเป็นเส้นขนานกับเส้นที่ต้องการหา และจะถูกเพิ่มค่าออฟเซ็ตในแนวแกน x และจะลดค่าออฟเซ็ตในแนวแกน Y

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.29 แสดงค่าออฟเซ็ตในแนวแกน X และ Y ที่เกิดจากเส้นทางเดินและเครื่องมือ



ค่าออฟเซ็ตในแนวแกน X และ แนวแกน Y

ค่าออฟเซ็ตจะเป็นค่าที่ใช้บวกเพิ่ม หรือ ลบออกจากแนวเส้นรอบรูปวงปิด เพื่อให้เครื่องมือเดินอยู่ภายนอก หรือ ภายในเส้นรอบรูป แล้วไม่กินเนื้องานของเส้นรอบรูป จากรูปที่ 4.29 จะได้สมการของเส้นออฟเซ็ตดังนี้

$$x = \frac{R_{tool}}{\sin(\theta)}$$

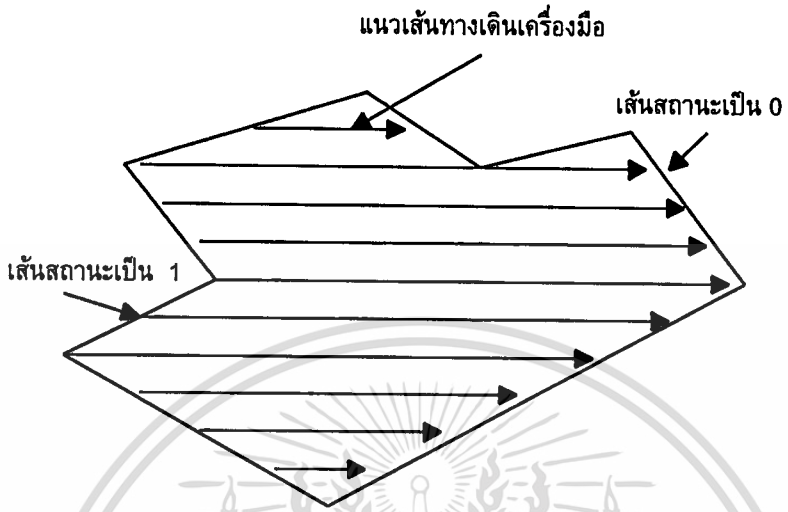
$$y = \frac{R_{tool}}{\cos(\theta)}$$

การหาเส้นทางเดินของเครื่องมือภายในเส้นรอบรูปวงปิด

เส้นทางเดินภายในเส้นรอบรูปจะกำหนดเป็นการเดินเส้นเป็นลำดับจากบนลงล่างในแนวแกน Y และในแต่ละลำดับของแนวแกน y จะเดินเป็นเส้นขนานกับแกน x ที่อยู่ภายในเส้นรอบรูปวงปิด ดังรูปที่ 4.30

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.30 แสดงแนวเส้นทางเดินของแนวกัดภายในเส้นรอบรูปวงปิด



เส้นทางเดินจะลากเป็นเส้นในแนวระดับจากซ้ายไปขวา และจะกัดในแนวเส้นที่อยู่ในตำแหน่งเส้นที่มีสถานะจาก 1 ไปยังเส้นที่มีสถานะเป็น 0 ดังรูปที่ 4.30

4.4 โปรแกรมส่วนของการจำลองการทำงาน (Graphic Simulation)

โปรแกรมส่วนนี้ใช้สำหรับตรวจสอบคำสั่งของโปรแกรมที่เขียนขึ้น สำหรับสั่งงานเครื่องกัดแนวดิ่งซีเอ็นซีว่า ลำดับของการทำงานถูกต้องหรือไม่ และยังช่วยให้เห็นวัตถุหลังการกัดเสร็จแล้วก่อนทำการกัดจริง โดยโปรแกรมในส่วนนี้จะทำการกัดชิ้นงานให้ดูบนหน้าจอของเครื่องไมโครคอมพิวเตอร์ และยังเป็นการแสดงแนวการกัดชิ้นงานอีกด้วย หลังจากที่กัดชิ้นงานเสร็จแล้วโปรแกรมจะแสดงภาพชิ้นงานทึบในลักษณะ 3 มิติ ให้ผู้ใช้งานได้ดูในมุมต่างๆ เพื่อให้เห็นภาพของวัตถุหลังการกัดชิ้นงานก่อนทำการกัดจริง

4.5 สรุป

ส่วนของซอฟท์แวร์นั้นได้ออกแบบให้อ่านจากโปรแกรมวาดแบบที่เก็บข้อมูลไฟล์แบบ dxf และสามารถกำหนดชั้นความลึกได้ และยังสามารถแสดงภาพ 3 มิติของชิ้นงานก่อนการกัดจริง ทำให้สะดวกต่อการใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

ตัวอย่างการใช้ซอฟท์แวร์

5.1 คำสั่งที่ใช้โดยทั่วไปในซอฟท์แวร์

โปรแกรมที่ทำการสร้างขึ้นมานี้จะใช้ระบบการทำงานแบบเมนูให้เลือกการทำงานสำหรับเลือกคำสั่ง โดยมีรายละเอียดของเมนูคำสั่งเป็นดังรูปที่ 5.1

รายละเอียดของเมนูแยกเป็นส่วนๆ ได้ ดังนี้

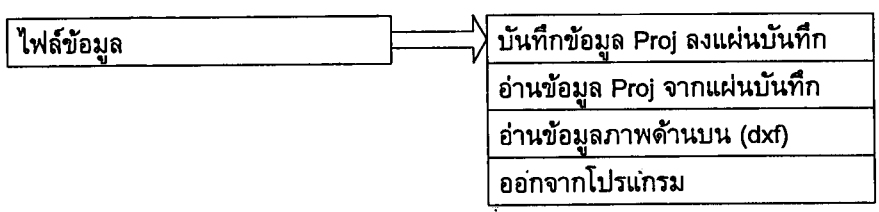
5.1.1 คำสั่งไฟล์

เป็นเมนูคำสั่งของการทำงานเกี่ยวกับไฟล์ข้อมูล เป็นส่วนที่ใช้บันทึก และเรียกไฟล์ข้อมูลกลับมาใช้งาน ไฟล์ที่ใช้งานในส่วนนี้ จะมีอยู่ 2 ชนิด คือ โปรเจคไฟล์เก็บข้อมูลรายละเอียดชิ้นงานที่จะทำ และ ไฟล์รูปภาพจากโปรแกรมวาดแบบ

โปรเจคไฟล์ นี้จะเก็บรายละเอียดของข้อมูลชิ้นงานที่จะสร้างขึ้น โดยรายละเอียดของไฟล์โปรเจคเป็นดังนี้

-มีชื่อไฟล์รูปภาพ ข้อมูลตัวอักษร รูปแบบ dxf
-ข้อมูลขนาดของชิ้นงาน กว้าง ยาว หนา
-ข้อมูลชั้นความลึก
-ข้อมูลของเครื่องมือที่ใช้กัด
-ชื่อไฟล์ของโปรแกรมควบคุมเครื่องจักร (ไฟล์นามสกุล iso)

รูปที่ 5.1 แสดงเมนูการทำงานไฟล์ข้อมูล

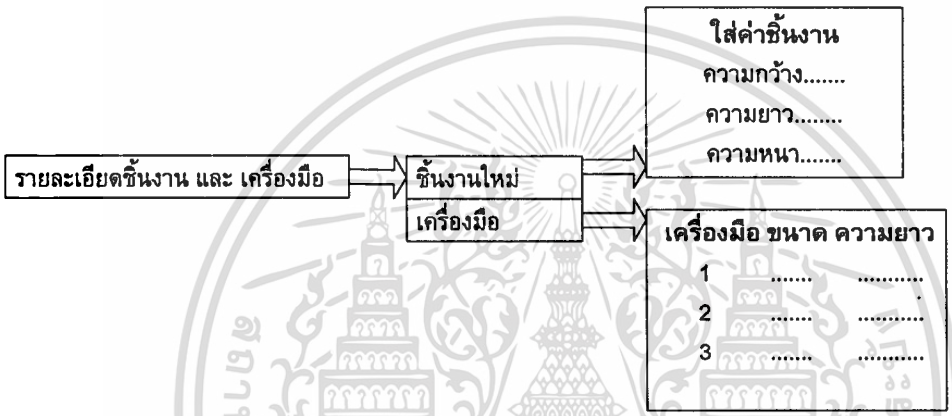


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.1.2 คำสั่งรายละเอียดชิ้นงาน และ เครื่องมือ

เป็นคำสั่งที่ใช้สำหรับป้อนข้อมูลรายละเอียดชิ้นงาน ว่ามีขนาดเท่าไร กว้าง ยาว และหนาเท่าไร และเป็นคำสั่งที่ใช้กำหนดเครื่องมือที่ใช้ในการกัด ว่าใช้เครื่องมือแบบไหน มีเส้นผ่าศูนย์กลางเท่าไร ความยาวเท่าไร จะได้ทำการสร้างโปรแกรมสั่งงานเครื่องจักรได้ถูก เมนูรายละเอียดชิ้นงาน และ เครื่องมือ แสดงในรูปที่ 5.2

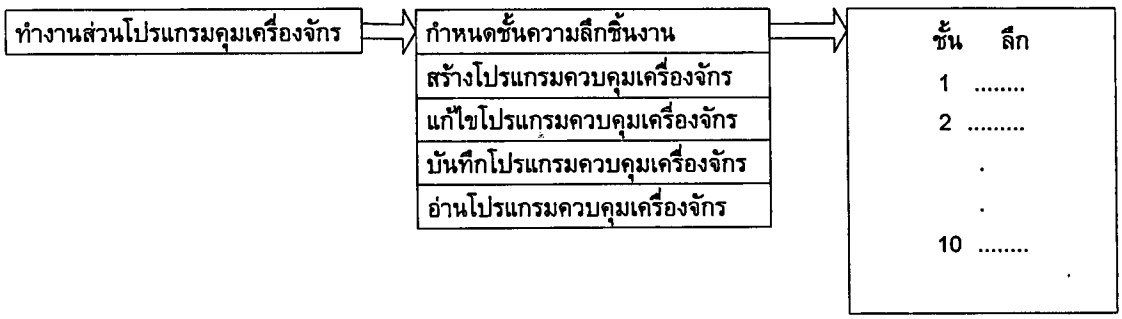
รูปที่ 5.2 แสดงเมนูรายละเอียดชิ้นงาน และ เครื่องมือ



5.1.3 คำสั่งทำงานส่วนโปรแกรมควบคุมเครื่องจักร

ในส่วนนี้จะเป็นการสร้างไฟล์สำหรับควบคุมเครื่องจักร โดยต้องทำการสร้างข้อมูลชิ้นงาน เครื่องมือ พร้อมทั้งรูปด้านบน(จากไฟล์โปรแกรมวาดแบบ)ก่อน

รูปที่ 5.3 แสดงเมนูทำงานส่วนโปรแกรมคุมเครื่องจักร

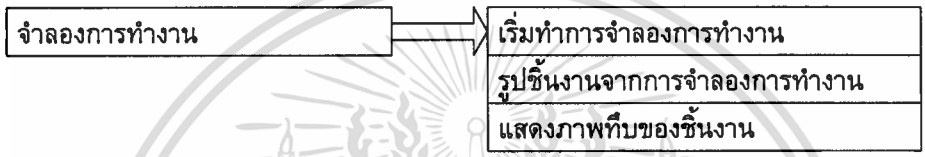


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.1.4 คำสั่งจำลองการทำงาน

ในส่วนนี้เป็นการเลือกที่จะให้เครื่องทำการจำลองการทำงานของเครื่องจักร โดยใช้คำสั่งควบคุมเครื่องจักร ที่โปรแกรมสร้างขึ้นมา หรือ ที่โหลดจากแผ่นบันทึกมา แล้วทำการจำลองการเคลื่อนที่ของเครื่องเพื่อใช้ทดสอบคำสั่งการทำงานว่าถูกต้องตามต้องการหรือไม่ พร้อมทั้งยังแสดงรูปของชิ้นงานหลังการกัดแล้วให้ดูได้ด้วย

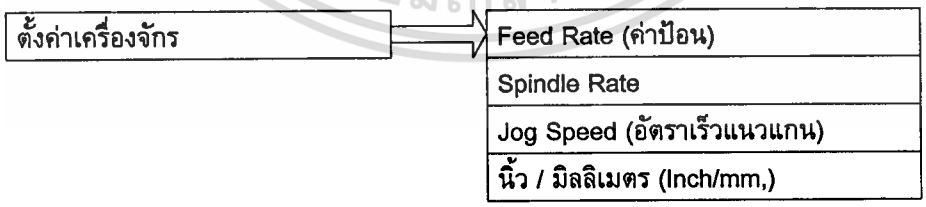
รูปที่ 5.4 แสดงคำสั่งจำลองการทำงาน



5.1.5 คำสั่งตั้งค่าเครื่องจักร

คำสั่งส่วนนี้ใช้สำหรับตั้งค่าต่างๆของเครื่องจักร ดังนี้ ตั้งค่าป้อน (Reed Rate) ตั้งความเร็วรอบหัวหมุน (Spindle Rate) ตั้งอัตราเร็วแนวแกน (Jog Speed) และ การเปลี่ยนหน่วยระหว่างนิ้ว กับ มิลลิเมตร

รูปที่ 5.5 แสดงคำสั่งสำหรับตั้งค่า

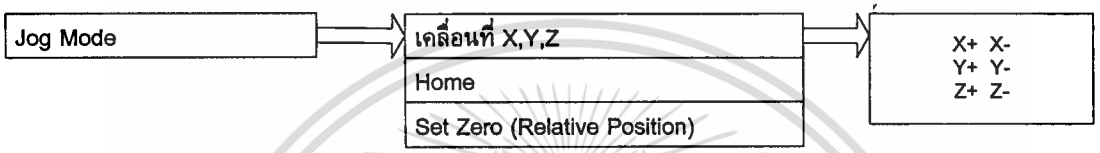


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.1.6 คำสั่ง Jog Mode

เป็นคำสั่งสำหรับให้เครื่องจักรทำงานตามที่ช่างควบคุม ควบคุมเครื่องจักรด้วยตัวเอง เหมือนๆ กับสั่งงานด้วยมือ เลื่อนแกน X,Y และ Z ทีละแกน มีการเลื่อนเข้าสู่ตำแหน่งเริ่มต้น (Home) และ มีการเซ็ตค่าตำแหน่งแบบอ้างอิงให้เป็นศูนย์ได้ (Set Zero)

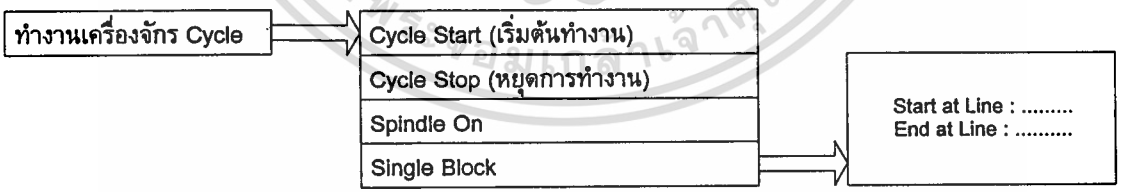
รูปที่ 5.6 แสดงคำสั่ง Jog Mode



5.1.7 คำสั่งทำงานเครื่องจักร (Cycle)

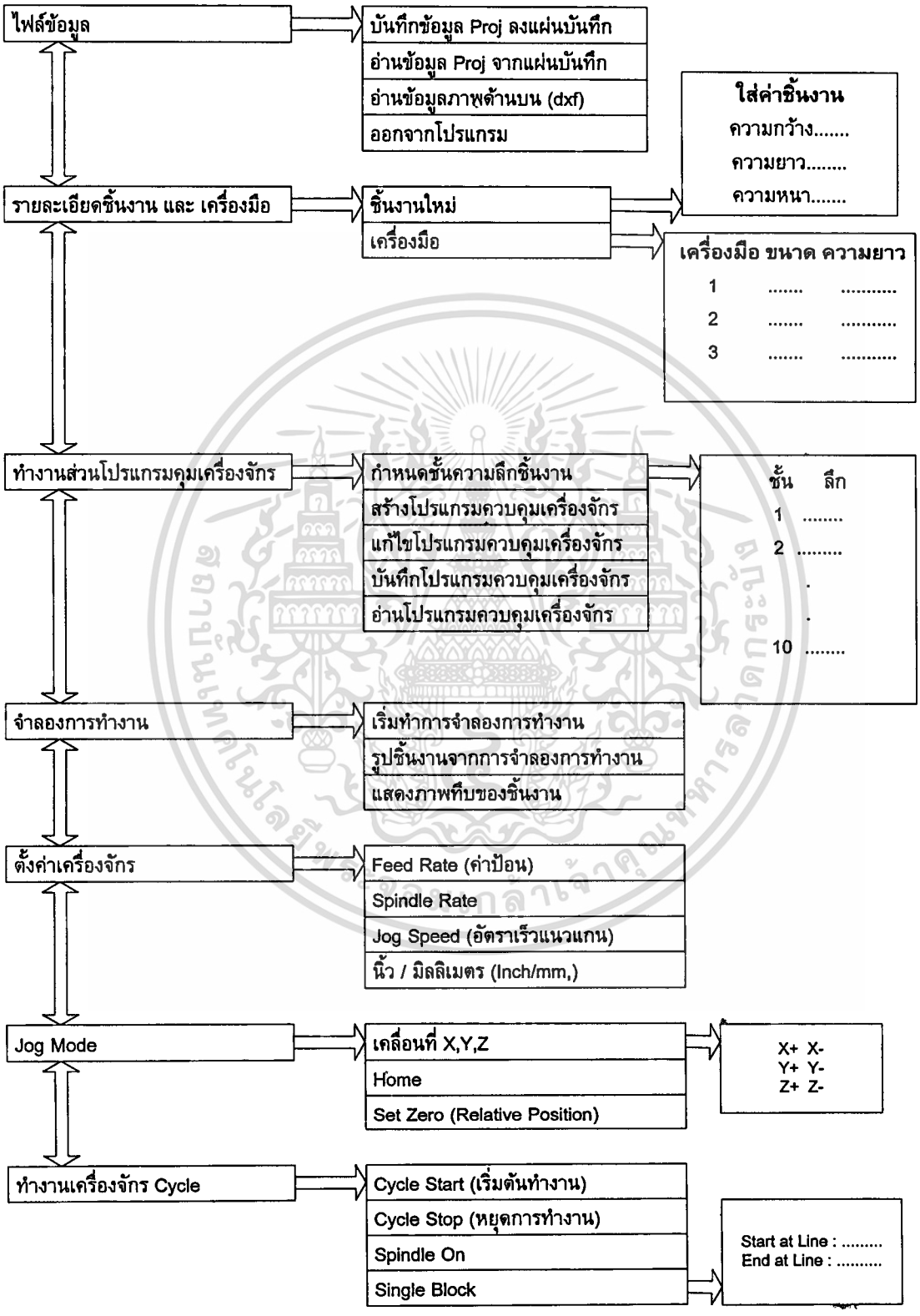
เป็นคำสั่งที่ใช้ให้เครื่องจักรทำงานตามโปรแกรมสั่งงานเครื่องจักรที่ได้สร้างเป็นโปรแกรม(G-code) ไว้แล้ว โดยสามารถทำงานเฉพาะส่วนของโปรแกรมได้ด้วย

รูปที่ 5.7 แสดงคำสั่งทำงานเครื่องจักร



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 5.8 แสดงเมนูการทำงานของโปรแกรม



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

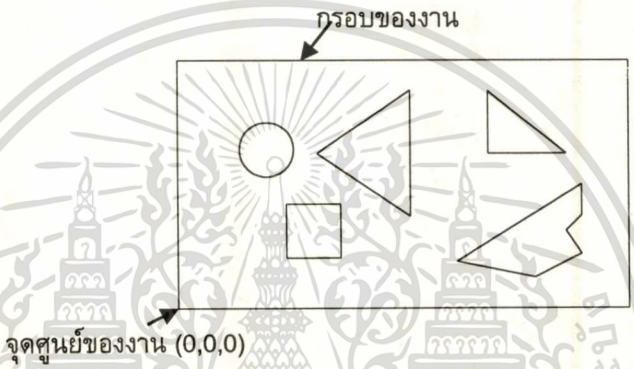
5.2 วิธีใช้งานโปรแกรม

การใช้งานโปรแกรมจะมีขั้นตอนการใช้งานดังนี้

1 เตรียมข้อมูลภาพด้านบนของงานจากโปรแกรมวาดแบบ ในที่นี้ใช้โปรแกรม Acad แล้วเก็บข้อมูลไว้ในไฟล์ (Dxf ของ Acad R.12) ดังตัวอย่างรูป 5.9 โดยมีข้อกำหนดของข้อมูลดังนี้คือ -ต้องมีกรอบเป็นรูปสี่เหลี่ยมที่มีขนาดเท่ากับขนาดของงานที่จะทำการกัด

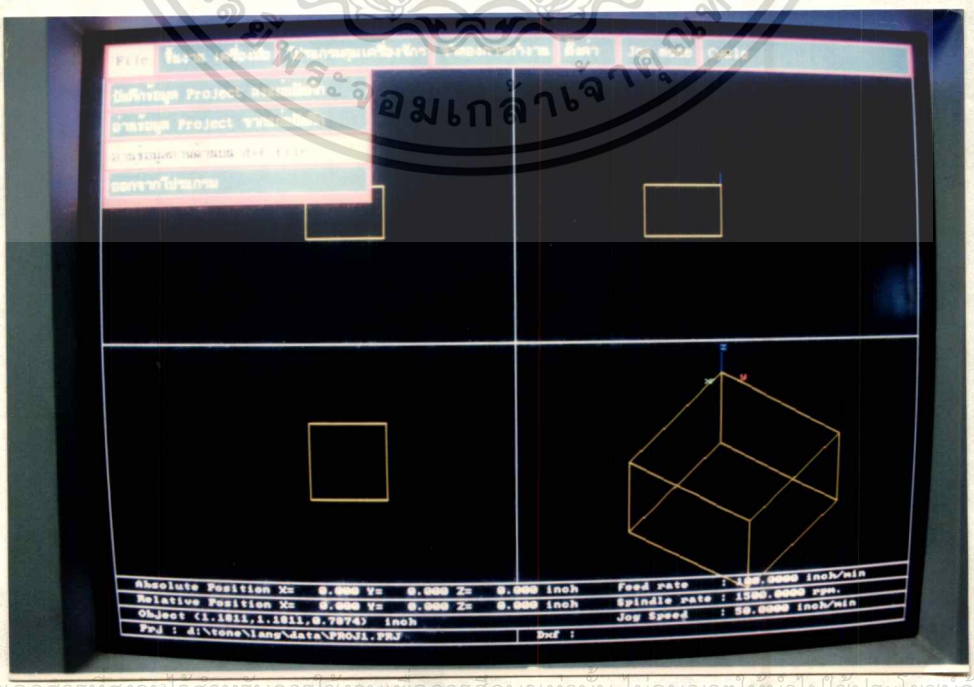
-กำหนดให้ด้านซ้ายล่างของกรอบสี่เหลี่ยมเป็นจุด (0,0,0) ของชิ้นงาน

รูปที่ 5.9 แสดงภาพชิ้นงานที่สร้างขึ้นเป็น Dxf ไฟล์ของ Acad R.12



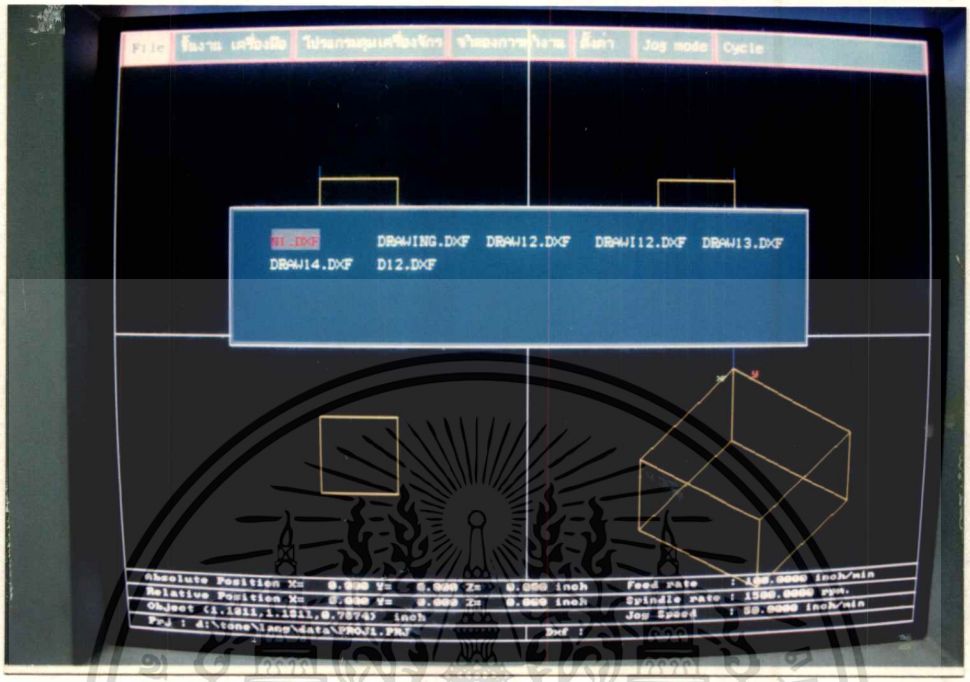
2 โหลดไฟล์ข้อมูลภาพ Dxf ที่เตรียมไว้เข้าสู่โปรแกรม โดยการใช้คำสั่งไฟล์ข้อมูล และเลือกอ่านข้อมูลภาพด้านบน แล้วเลือกไฟล์ข้อมูล Dxf ที่ต้องการ ดังรูปที่ 5.2

ภาพที่ 5.10 แสดงการใช้คำสั่งไฟล์ และเลือกอ่านข้อมูลภาพด้านบน Dxf file

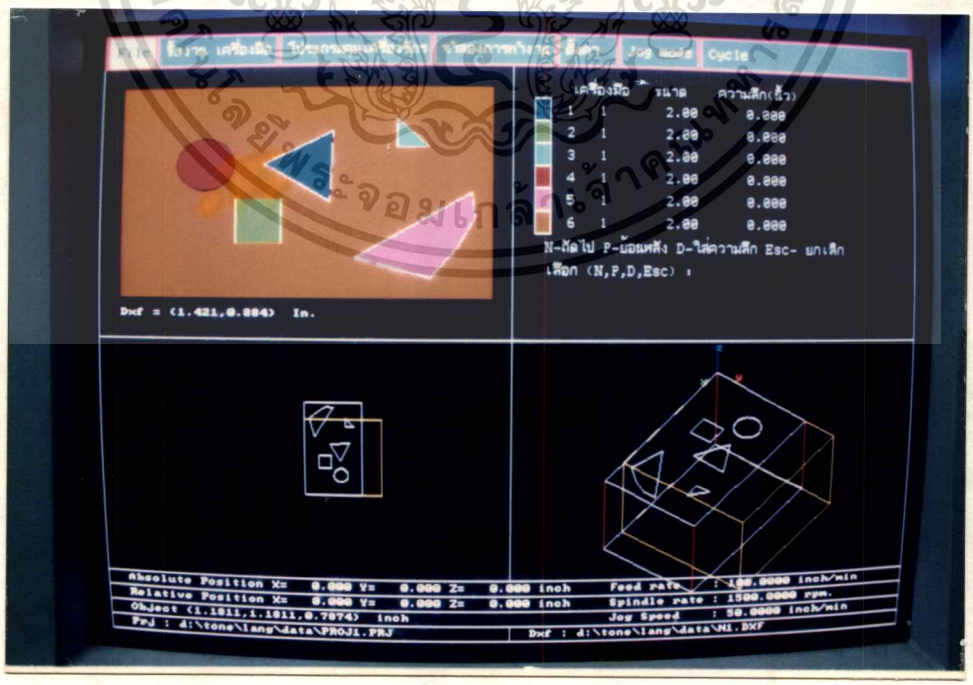


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ 5.11 แสดงการเลือกไฟล์ Dxf ที่ต้องการ



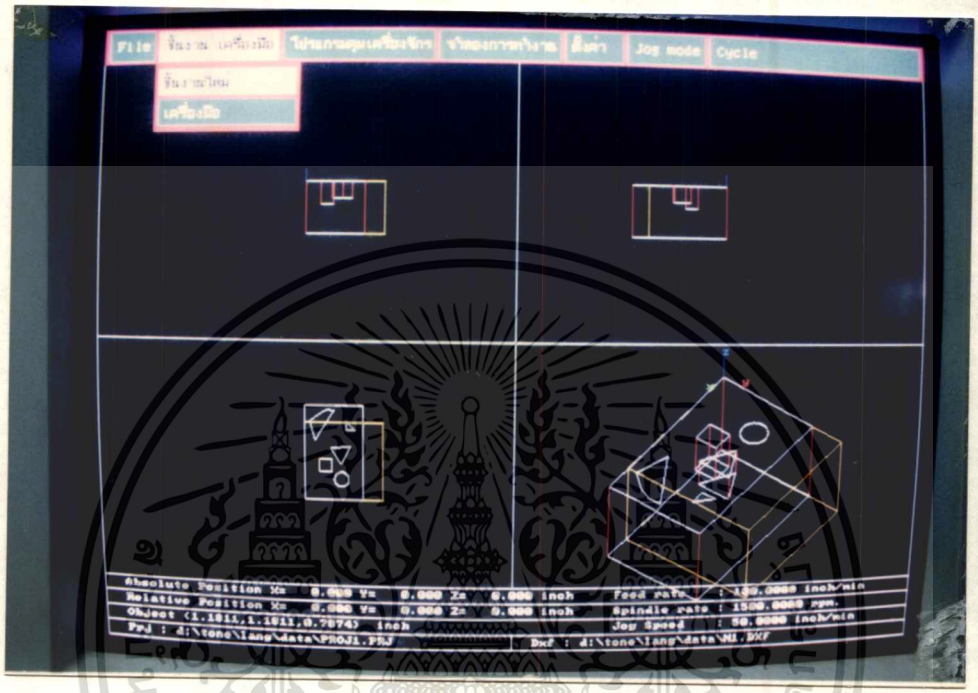
ภาพที่ 5.12 แสดงการโหลดไฟล์ Dxf เข้าสู่โปรแกรมแล้ว



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3 กำหนดขนาดวัตถุที่ใช้ในการกัด โดยการใช้คำสั่งชิ้นงานเครื่องมือ และ เลือกชิ้นงานใหม่ แล้วใส่ค่าความกว้าง ความยาว และ ความหนาของชิ้นงานเข้าไป ดังภาพที่ 5.13

ภาพที่ 5.13 การใช้คำสั่งชิ้นงานเครื่องมือ และเลือกชิ้นงานใหม่

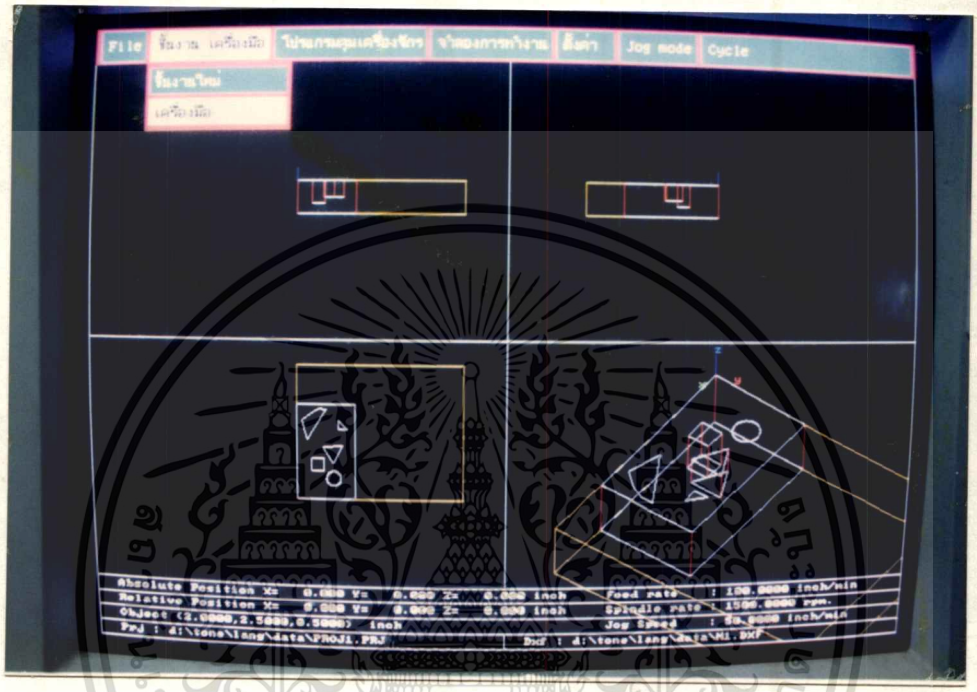


ภาพที่ 5.14 แสดงการกำหนดขนาดของชิ้นงานที่ใช้ในการกัดในโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4 ใส่ค่าเครื่องมือ โดยการกำหนดขนาด และความยาวของเครื่องมือที่ใช้ในการทำงาน เลือกคำสั่งชิ้นงาน เครื่องมือ และเลือกเมนูเครื่องมือ แล้วทำการกำหนดขนาดเครื่องมือที่มีอยู่ ดังภาพที่ 5.15 และ 5.16

ภาพที่ 5.15 แสดงการใช้คำสั่งชิ้นงาน เครื่องมือ และเลือกเครื่องมือ

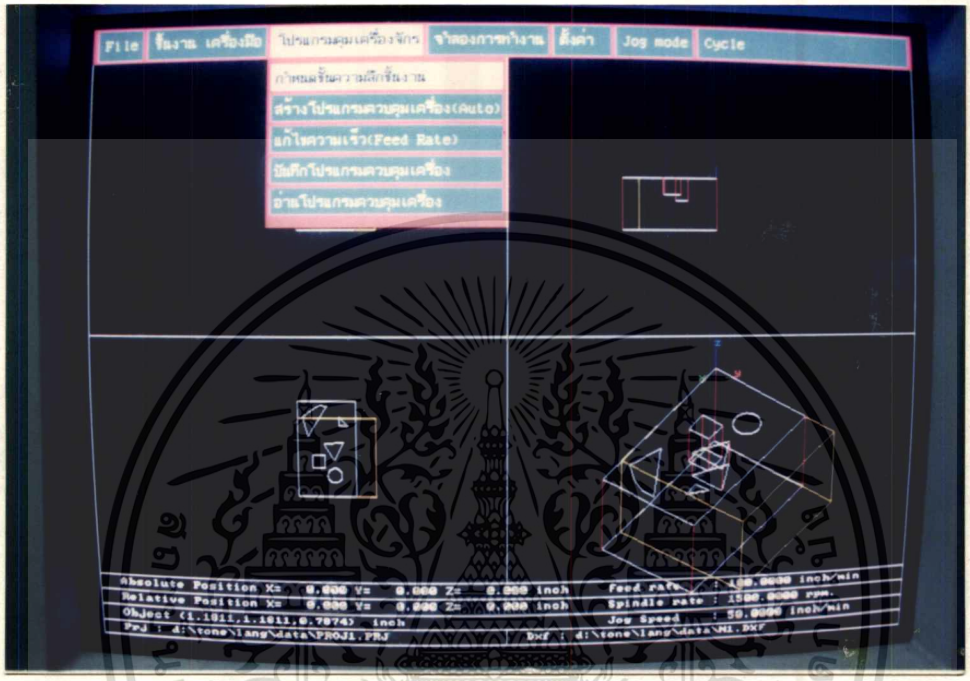


ภาพที่ 5.16 แสดงการใส่ค่าเครื่องมือที่ใช้งานลงไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5 กำหนดชั้นความลึกชิ้นงาน โดยเลือกคำสั่งส่วนการทำงานโปรแกรมควบคุมเครื่อง แล้วเลือกเมนูกำหนดชั้นความลึกชิ้นงาน แล้วใส่ค่าความลึกชิ้นงาน ดังภาพที่ 5.17 และ 5.18

ภาพที่ 5.17 แสดงการใช้คำสั่งโปรแกรมเครื่องจักร และเลือกกำหนดชั้นความลึกชิ้นงาน



ภาพที่ 5.18 แสดงการใส่ค่าชั้นความลึกชิ้นงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6 สั่งให้โปรแกรมสร้างโปรแกรมควบคุมเครื่องจักรขึ้น โดยการเลือกคำสั่งส่วนการทำงานเครื่องจักร แล้วเลือกเมนูสร้างโปรแกรมควบคุมเครื่องจักร ดังภาพที่ 5.19 โปรแกรมจะนำข้อมูลที่ใส่ให้ทั้งหมดไปทำการคำนวณเพื่อหาเส้นทางเดินของหัวกัด

ภาพที่ 5.19 การใช้คำสั่งโปรแกรมควบคุมเครื่องจักร และเลือกสร้างโปรแกรมควบคุมเครื่อง

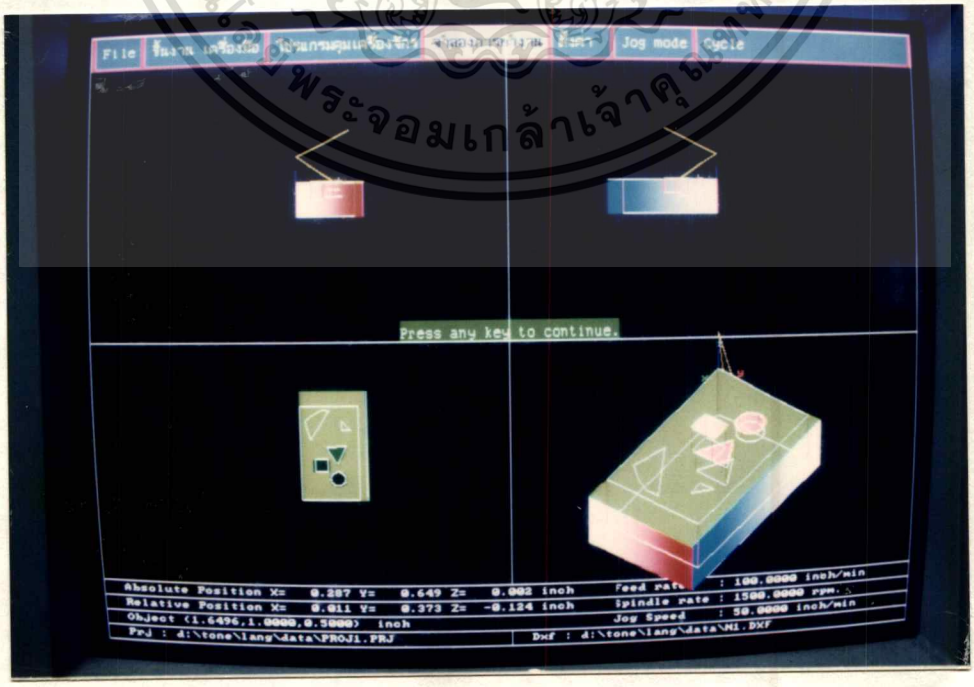
ภาพที่ 5.20 แสดงหน้าจอหลังจากการคำนวณหาเส้นทางเดินการกัดแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7 ทดสอบโปรแกรมทำงานเครื่องจักร โดยการเลือกเมนูจำลอง แล้วเลือกจำลองการทำงานดังภาพที่ 5.21

ภาพที่ 5.21 แสดงการเลือกเมนูจำลองการทำงาน แล้วแสดงการจำลองการทำงาน

ภาพที่ 5.22 แสดงชิ้นงานจำลองหลังการกัดแล้ว

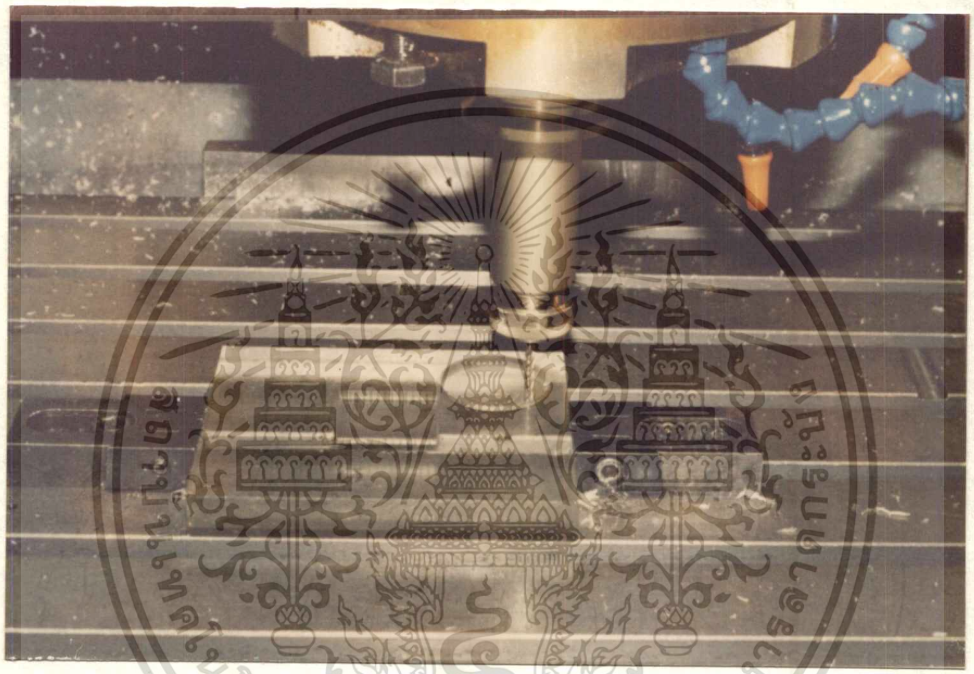


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8 ทำการกัดชิ้นงาน โดยชิ้นงานและตัวจับยึดชิ้นงานจะต้องมีข้อกำหนดดังนี้

-ชิ้นงานจะเป็นรูปสี่เหลี่ยมโดยมีขนาดของด้านต่างๆ เท่าๆกับขนาดที่กำหนดไว้ในโปรแกรม ส่วนที่ใช้จับชิ้นงานควรมีความสูงน้อยกว่าส่วนของความหนาของชิ้นงาน จะได้กัดชิ้นงานในแนวลึกได้ ดังภาพที่ 5.23

ภาพที่ 5.23 แสดงการจับยึดวัตถุที่จะกัด และ แสดงการกัด



สรุปขั้นตอนในการใช้ซอฟท์แวร์ได้ดังนี้

1 วาดแบบด้านบนชิ้นงานในโปรแกรมวาดแบบก่อนเก็บในไฟล์ dxf ของ Acad R.12
2 ใช้โปรแกรมที่ทำขึ้นโหลดข้อมูลภาพ dxf เก็บไว้ในโปรแกรมโดยการเลือกเมนูไฟล์
3 กำหนดค่าพารามิเตอร์ต่างๆ ความเร็วในการกัด ขนาดดอกกัด รายละเอียดชิ้นงาน
4 สั่งให้โปรแกรมทำการสร้างโค้ดควบคุมเครื่องจักร
5 จำลองการทำงานเพื่อดูภาพชิ้นงานที่จะทำการกัดจากโปรแกรม
6 สั่งเครื่องทำการกัดชิ้นงาน

หลังจากทำตามขั้นตอนข้างต้นแล้วก็จะได้ชิ้นงานตามต้องการเพื่อนำไปใช้งานต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6
การทดลอง สรุป และวิจารณ์ผลการทดลอง

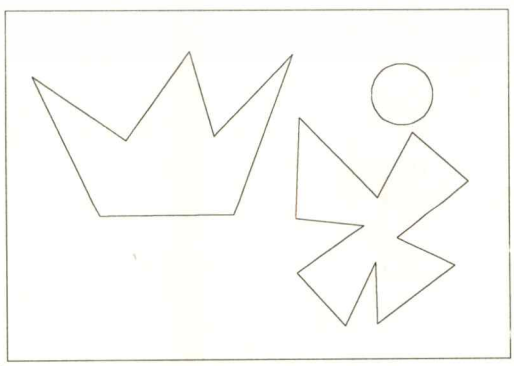
6.1 การทดลอง

การทดลองเป็นการทดสอบฮาร์ดแวร์ และ ซอฟท์แวร์ที่สร้างขึ้นว่าสามารถทำงานตามแบบของเครื่องกัดซีเอ็นซีโดยทั่วไปได้ มีการทดสอบส่วนของการจำลองการทำงานเพื่อดูชิ้นงานสำเร็จก่อนการกัดจริง และทดสอบการสร้างโค้ดควบคุมเครื่องจักรจากไฟล์รูปภาพ dxf โดยนำโค้ดควบคุมเครื่องกัดที่ได้จากโปรแกรมไปสั่งงานผ่านการ์ดควบคุมที่สร้างขึ้น เพื่อควบคุมการทำงานของเครื่องกัดแนวดิ่งซีเอ็นซีต้นแบบ

6.1.1 การทดลองส่วนการจำลองการทำงาน

ทดลองโดยการสร้างแบบงานขึ้นจากโปรแกรม Acad ตามรูปที่ 6.1 แล้วโหลดรูปงานเข้าสู่โปรแกรมในโครงงาน จากนั้นได้กำหนดชั้นความลึกตามตารางที่ 6.1 หลังจากนั้นสั่งให้โปรแกรมสร้างโค้ดควบคุมเครื่องกัด เมื่อโปรแกรมได้สร้างโค้ดควาบคุมเครื่องกัดเสร็จแล้ว สั่งโปรแกรมจำลองการทำงานเครื่องกัด จะได้ผลการทดลองเป็นรูปชิ้นงานสำเร็จหลังการกัดดังภาพที่ 6.2

รูปที่ 6.1 ภาพงานที่ใช้ทดสอบ



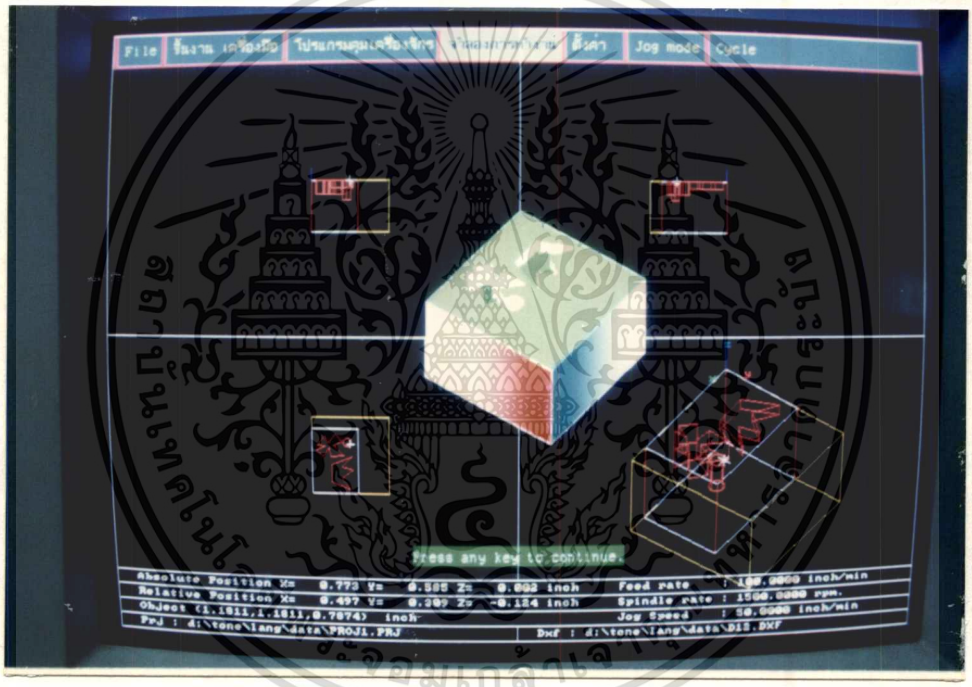
จุดศูนย์ชิ้นงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 6.1 การกำหนดชั้นความลึกงานกัด

ชั้นที่	ลึก (mm.)
1	2
2	4
3	6

ภาพที่ 6.2 แสดงชิ้นงานจำลองหลังการกัดจากคำสั่งจำลองการทำงาน

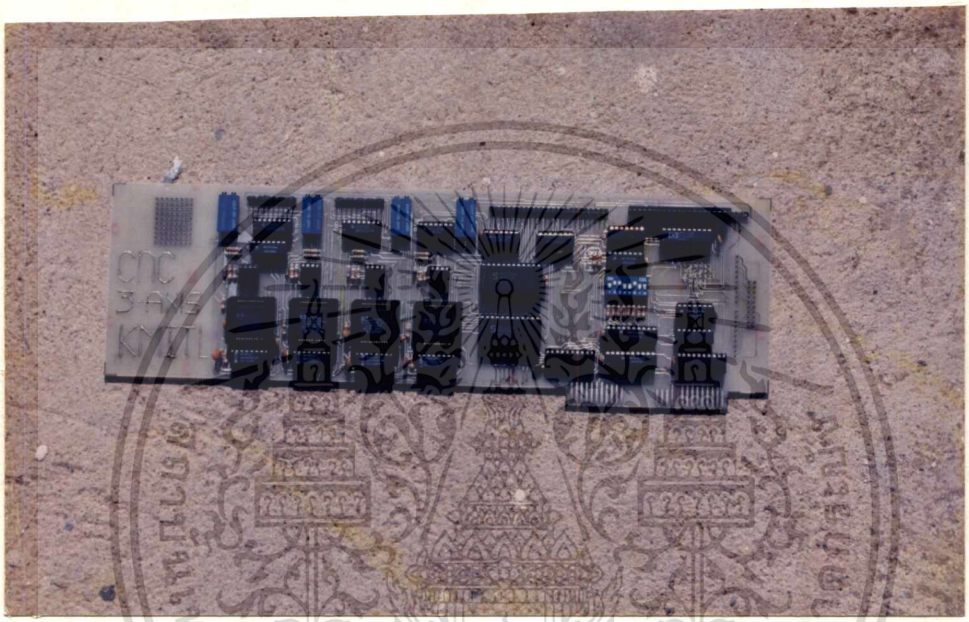


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

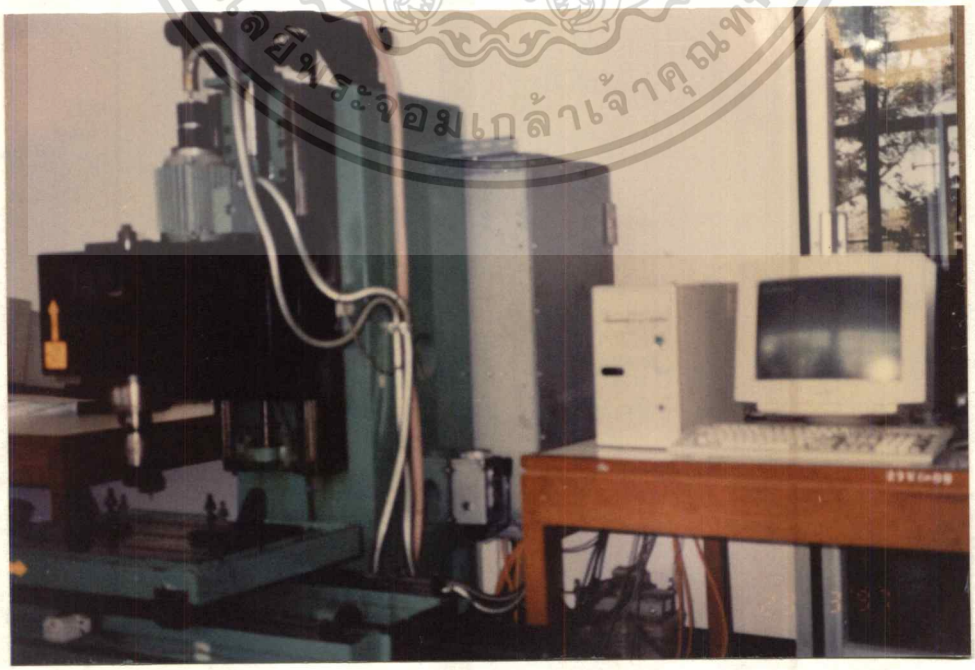
6.1.2 ทดลองกัดงานโดยการสั่งผ่านการ์ดควบคุมการทำงาน

ทำการทดลองโดยการนำการ์ดควบคุมมอเตอร์ที่สร้างขึ้นในภาพที่ 6.3 ไปสั่งงานเครื่องกัดแนวดิ่งซีเอ็นซีต้นแบบในภาพที่ 6.4 แล้วทดลองเขียนโค้ดโปรแกรมควบคุมเครื่องจักร แล้วสั่งกัดชิ้นงาน (cycle start) โดยมีการทดลองดังนี้

ภาพที่ 6.3 แสดงต้นแบบส่วนควบคุมมอเตอร์ที่สร้างขึ้นในโครงงาน



ภาพที่ 6.4 แสดงเครื่องกัดแนวดิ่งซีเอ็นซีที่ใช้ในการทดลอง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทดลองกัดเป็นวงกลมตามโค้ดที่กำหนดดังนี้

```
N1 G00 X0.000 Y0.000 Z10.000
N2 M03 S1500
N3 G00 X-40.000
N4 G01 Z-3.000 F100
N5 G02 X40.000 Y0.000 R40.000 F100
N6 G02 X-40.000 Y0.000 R40.000 F100
N7 G00 Z50.000
N8 G00 X0.000 Y0.000
N9 M02
```

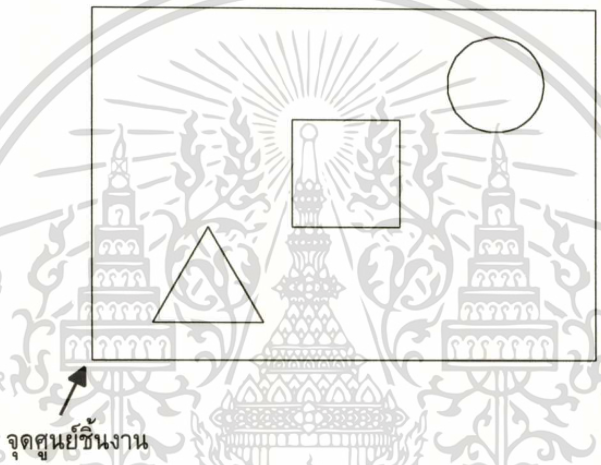
ได้ผลดังนี้

ภาพที่ 6.5 แสดงผลจากการสั่งทำงานเครื่องจักรกัดเป็นรูปวงกลม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทดลองกัดงานเป็นรูปสี่เหลี่ยม สามเหลี่ยม และวงกลม โดยการสร้างรูปชิ้นงานในโปรแกรมวาดแบบดังในรูปที่ 6.6 แล้วกำหนดชั้นความลึกในตารางที่ 6.2 แล้วจึงสั่งให้โปรแกรมสร้างโค้ดสั่งงานเครื่องจักร และสั่งให้เครื่องจักรทำงานตามโค้ดดังกล่าว พร้อมทั้งจำลองการทำงานได้ผลดังนี้

รูปที่ 6.6 แสดงรูปงานจากโปรแกรมวาดแบบ



ตารางที่ 6.2 แสดงชั้นความลึกของชิ้นงานทดลอง

รูป	ค่าความลึก(มม.)
วงกลม	7
สี่เหลี่ยม	5
สามเหลี่ยม	3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โค้ดควบคุมเครื่องจักรที่ได้จากโปรแกรมในโครงงาน

```
N10 G00 X0 Y0 Z100
N15 M03 S1500
N20 G00 X128.00 Y112.000 Z5.000
N25 G01 Z-5.000 F100
N30 G01 X128.000 Y68.000 F100
N35 G01 Z5.000 F500
N40 G00 X92.000 Y112.000
N45 G01 Z-5.000 F100
N50 G01 X128.000 Y112.000 F100
N55 G01 Z5.000 F500
N60 G00 X92.000 Y68.000
N65 G01 Z-5.000 F100
N70 G01 X92.000 Y112.000 F100
N75 G01 Z5.000 F500
N80 G00 X128.000 Y68.000
N85 G01 Z-5.000 F100
N90 G01 X92.000 Y68.000 F100
N95 G01 Z5.000 F500
N100 G00 X92.000 Y68.000
N105 G01 Z-5.000 F100
N110 G01 X128.000 Y68.000 F100
N115 G01 Z5.000 F500
N120 G00 X92.000 Y70.000
N125 G01 Z-5.000 F100
N130 G01 X128.000 Y70.000 F100
N135 G01 Z5.000 F500
N140 G00 X92.000 Y72.000
N145 G01 Z-5.000 F100
N150 G01 X128.000 Y72.000 F100
N155 G01 Z5.000 F500
N160 G00 X92.000 Y74.000
N165 G01 Z-5.000 F100
N170 G01 X128.000 Y74.000 F100
N175 G01 Z5.000 F500
N180 G00 X92.000 Y76.000
N185 G01 Z-5.000 F100
N190 G01 X128.000 Y76.000 F100
N195 G01 Z5.000 F500
N200 G00 X92.000 Y78.000
N205 G01 Z-5.000 F100
N210 G01 X128.000 Y78.000 F100
N215 G01 Z5.000 F500
N220 G00 X92.000 Y80.000
N225 G01 Z-5.000 F100
N230 G01 X128.000 Y80.000 F100
N235 G01 Z5.000 F500
N240 G00 X92.000 Y82.000
N245 G01 Z-5.000 F100
N250 G01 X128.000 Y82.000 F100
N255 G01 Z5.000 F500
N260 G00 X92.000 Y84.000
N265 G01 Z-5.000 F100
N270 G01 X128.000 Y84.000 F100
N275 G01 Z5.000 F500
N280 G00 X92.000 Y86.000
N285 G01 Z-5.000 F100
N290 G01 X128.000 Y86.000 F100
N295 G01 Z5.000 F500
N300 G00 X92.000 Y88.000
N305 G01 Z-5.000 F100
N310 G01 X128.000 Y88.000 F100
N315 G01 Z5.000 F500
N320 G00 X92.000 Y90.000
N325 G01 Z-5.000 F100
N330 G01 X128.000 Y90.000 F100
N335 G01 Z5.000 F500
N340 G00 X92.000 Y92.000
N345 G01 Z-5.000 F100
N350 G01 X128.000 Y92.000 F100
N355 G01 Z5.000 F500
N360 G00 X92.000 Y94.000
N365 G01 Z-5.000 F100
N370 G01 X128.000 Y94.000 F100
N375 G01 Z5.000 F500
N380 G00 X92.000 Y96.000
N385 G01 Z-5.000 F100
N390 G01 X128.000 Y96.000 F100
N395 G01 Z5.000 F500
N400 G00 X92.000 Y98.000
N405 G01 Z-5.000 F100
N410 G01 X128.000 Y98.000 F100
N415 G01 Z5.000 F500
N420 G00 X92.000 Y100.000
N425 G01 Z-5.000 F100
N430 G01 X128.000 Y100.000 F100
N435 G01 Z5.000 F500
N440 G00 X92.000 Y102.000
N445 G01 Z-5.000 F100
N450 G01 X128.000 Y102.000 F100
N455 G01 Z5.000 F500
N460 G00 X92.000 Y104.000
N465 G01 Z-5.000 F100
N470 G01 X128.000 Y104.000 F100
N475 G01 Z5.000 F500
N480 G00 X92.000 Y106.000
N485 G01 Z-5.000 F100
N490 G01 X128.000 Y106.000 F100
N495 G01 Z5.000 F500
N500 G00 X92.000 Y108.000
N505 G01 Z-5.000 F100
N510 G01 X128.000 Y108.000 F100
N515 G01 Z5.000 F500
N520 G00 X92.000 Y110.000
N525 G01 Z-5.000 F100
N530 G01 X128.000 Y110.000 F100
N535 G01 Z5.000 F500
N540 G00 X-10.000 Y0.000
N545 G01 Z-0.000 F100
N550 G01 X0.000 Y0.000 F100
N555 G01 Z5.000 F500
N560 G00 X140.000 Y130.000
N565 G01 Z-7.000 F100
N570 G02 X140.000 Y130.000
I20.000 J0.000 F100
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

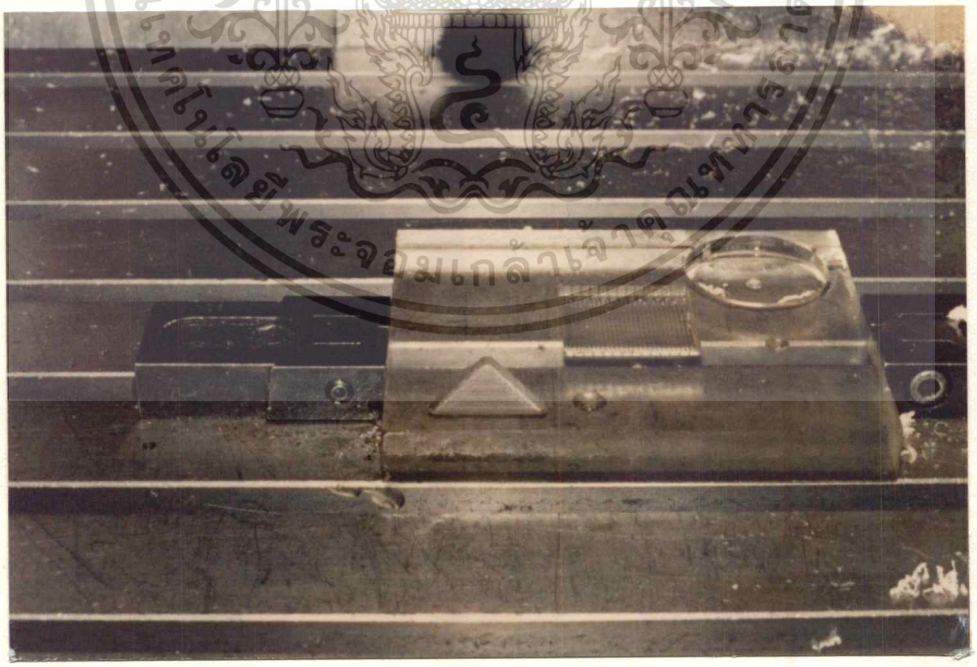
```
N575 G01 Z5.000 F500
N580 G00 X160.000 Y112.000
N585 G01 Z-7.000 F100
N590 G01 X160.000 Y112.000 F100
N595 G01 Z5.000 F500
N600 G00 X168.246 Y114.000
N605 G01 Z-7.000 F100
N610 G01 X151.754 Y114.000 F100
N615 G01 Z5.000 F500
N620 G00 X171.314 Y116.000
N625 G01 Z-7.000 F100
N630 G01 X148.686 Y116.000 F100
N635 G01 Z5.000 F500
N640 G00 X173.416 Y118.000
N645 G01 Z-7.000 F100
N650 G01 X146.584 Y118.000 F100
N655 G01 Z5.000 F500
N660 G00 X174.967 Y120.000
N665 G01 Z-7.000 F100
N670 G01 X145.033 Y120.000 F100
N675 G01 Z5.000 F500
N680 G00 X176.125 Y122.000
N685 G01 Z-7.000 F100
N690 G01 X143.875 Y122.000 F100
N695 G01 Z5.000 F500
N700 G00 X176.971 Y124.000
N705 G01 Z-7.000 F100
N710 G01 X143.029 Y124.000 F100
N715 G01 Z5.000 F500
N720 G00 X177.550 Y126.000
N725 G01 Z-7.000 F100
N730 G01 X142.450 Y126.000 F100
N735 G01 Z5.000 F500
N740 G00 X177.889 Y128.000
N745 G01 Z-7.000 F100
N750 G01 X142.111 Y128.000 F100
N755 G01 Z5.000 F500
N760 G00 X178.000 Y130.000
N765 G01 Z-7.000 F100
N770 G01 X142.000 Y130.000 F100
N775 G01 Z5.000 F500
N780 G00 X177.889 Y132.000
N785 G01 Z-7.000 F100
N790 G01 X142.111 Y132.000 F100
N795 G01 Z5.000 F500
N800 G00 X177.550 Y134.000
N805 G01 Z-7.000 F100
N810 G01 X142.450 Y134.000 F100
N815 G01 Z5.000 F500
N820 G00 X176.971 Y136.000
N825 G01 Z-7.000 F100
N830 G01 X143.029 Y136.000 F100
N835 G01 Z5.000 F500
N840 G00 X176.125 Y138.000
N845 G01 Z-7.000 F100
N850 G01 X143.875 Y138.000 F100
N855 G01 Z5.000 F500
N860 G00 X174.967 Y140.000
N865 G01 Z-7.000 F100
N870 G01 X145.033 Y140.000 F100
N875 G01 Z5.000 F500
N880 G00 X173.416 Y142.000
N885 G01 Z-7.000 F100
N890 G01 X146.584 Y142.000 F100
N895 G01 Z5.000 F500
N900 G00 X171.314 Y144.000
N905 G01 Z-7.000 F100
N910 G01 X148.686 Y144.000 F100
N915 G01 Z5.000 F500
N920 G00 X168.246 Y146.000
N925 G01 Z-7.000 F100
N930 G01 X151.754 Y146.000 F100
N935 G01 Z5.000 F500
N940 G00 X160.000 Y148.000
N945 G01 Z-7.000 F100
N950 G01 X160.000 Y148.000 F100
N955 G01 Z5.000 F500
N960 G00 X40.370 Y22.000
N965 G01 Z-3.000 F100
N970 G01 X60.000 Y56.000 F100
N975 G01 Z5.000 F500
N980 G00 X79.630 Y22.000
N985 G01 Z-3.000 F100
N990 G01 X40.370 Y22.000 F100
N995 G01 Z5.000 F500
N1000 G00 X60.000 Y56.000
N1005 G01 Z-3.000 F100
N1010 G01 X79.630 Y22.000 F100
N1015 G01 Z5.000 F500
N1020 G00 X-10.000 Y0.000
N1025 G01 Z-0.000 F100
N1030 G01 X0.000 Y0.000 F100
N1035 G01 Z5.000 F500
N1040 G00 X41.525 Y24.000
N1045 G01 Z-3.000 F100
N1050 G01 X78.475 Y24.000 F100
N1055 G01 Z5.000 F500
N1060 G00 X42.679 Y26.000
N1065 G01 Z-3.000 F100
N1070 G01 X77.321 Y26.000 F100
N1075 G01 Z5.000 F500
N1080 G00 X43.834 Y28.000
N1085 G01 Z-3.000 F100
N1090 G01 X76.166 Y28.000 F100
N1095 G01 Z5.000 F500
N1100 G00 X44.989 Y30.000
N1105 G01 Z-3.000 F100
N1110 G01 X75.011 Y30.000 F100
N1115 G01 Z5.000 F500
N1120 G00 X46.144 Y32.000
N1125 G01 Z-3.000 F100
N1130 G01 X73.856 Y32.000 F100
N1135 G01 Z5.000 F500
N1140 G00 X47.298 Y34.000
N1145 G01 Z-3.000 F100
N1150 G01 X72.702 Y34.000 F100
N1155 G01 Z5.000 F500
N1160 G00 X48.453 Y36.000
N1165 G01 Z-3.000 F100
N1170 G01 X71.547 Y36.000 F100
N1175 G01 Z5.000 F500
N1180 G00 X49.608 Y38.000
N1185 G01 Z-3.000 F100
N1190 G01 X70.392 Y38.000 F100
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
N1195 G01 Z5.000 F500
N1200 G00 X50.762 Y40.000
N1205 G01 Z-3.000 F100
N1210 G01 X69.238 Y40.000 F100
N1215 G01 Z5.000 F500
N1220 G00 X51.917 Y42.000
N1225 G01 Z-3.000 F100
N1230 G01 X68.083 Y42.000 F100
N1235 G01 Z5.000 F500
N1240 G00 X53.072 Y44.000
N1245 G01 Z-3.000 F100
N1250 G01 X66.928 Y44.000 F100
N1255 G01 Z5.000 F500
N1260 G00 X54.226 Y46.000
N1265 G01 Z-3.000 F100
N1270 G01 X65.774 Y46.000 F100
N1275 G01 Z5.000 F500
N1280 G00 X55.381 Y48.000
N1285 G01 Z-3.000 F100
N1290 G01 X64.619 Y48.000 F100
N1295 G01 Z5.000 F500
N1300 G00 X56.536 Y50.000
N1305 G01 Z-3.000 F100
N1310 G01 X63.464 Y50.000 F100
N1315 G01 Z5.000 F500
N1320 G00 X57.691 Y52.000
N1325 G01 Z-3.000 F100
N1330 G01 X62.309 Y52.000 F100
N1335 G01 Z5.000 F500
N1340 G00 X58.845 Y54.000
N1345 G01 Z-3.000 F100
N1350 G01 X61.155 Y54.000 F100
N1355 G01 Z5.000 F500
N1360 G00 X60.000 Y56.000
N1365 G01 Z-3.000 F100
N1370 G01 X60.000 Y56.000 F100
N1375 G01 Z5.000 F500
N1380 G00 X60.000 Y56.000
N1385 G01 Z-3.000 F100
N1390 G01 X60.000 Y56.000 F100
N1395 G01 Z0.500
N1400 M02
```

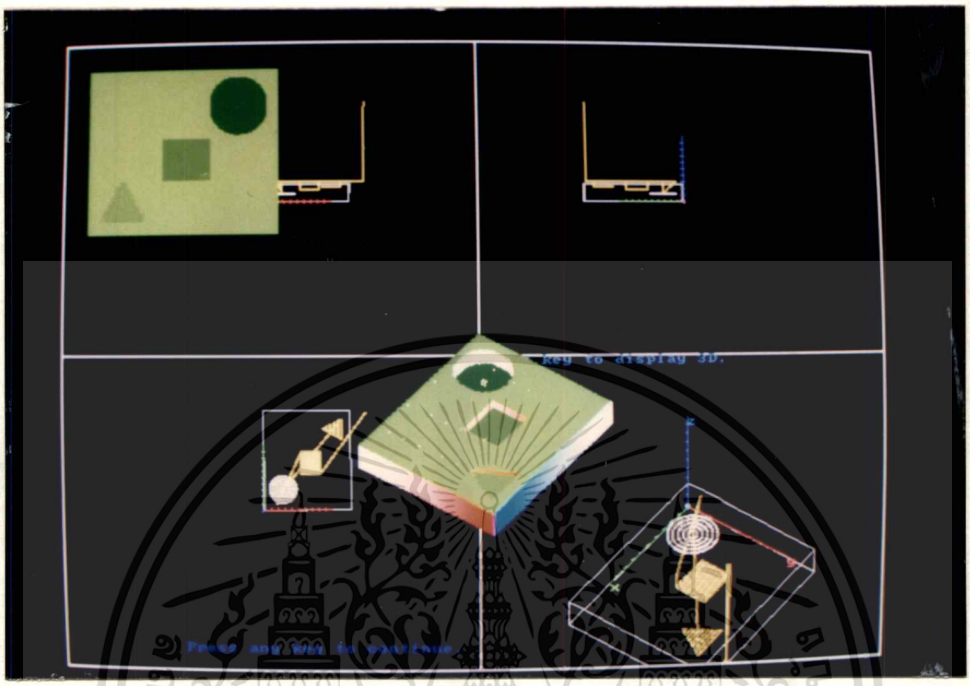
ได้ผลการทดลองดังนี้

ภาพที่ 6.7 แสดงชิ้นงานหลังจากการกัดโดยโค้ดจากโปรแกรมในโครงงาน



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ 6.8 แสองชิ้นงานหลังการกัดจากการจำลองการทำงาน



6.2 สรุปผลการทดลอง

6.2.1 ส่วนของฮาร์ดแวร์

ส่วนของฮาร์ดแวร์นั้นสามารถทำงานตามที่ต้องการได้เป็นอย่างดี การกำหนดค่าความละเอียดในการสั่งให้เคลื่อนที่ไปยังตำแหน่งที่กำหนด จะขึ้นอยู่กับค่าของจำนวนพัลของเอ็นโค้ดเดอร์ที่ใช้ และค่าความผิดพลาดบางส่วนที่เกิดจะมาจากการสั่นของหัวหมุน อาจเนื่องมาจากสายพานหย่อน หรือ การปรับค่าพารามิเตอร์ของตัวขับมอเตอร์ที่อาจสามารถปรับค่าได้ดีกว่านี้

6.2.2 ส่วนของซอฟท์แวร์

ส่วนของซอฟท์แวร์นั้นก็ใช้งานง่ายพอควร แต่ในบางครั้งอาจมีความไม่สะดวกบางประการในส่วนของการวาดแบบของชิ้นงานนั้นผู้ใช้ต้องมีความสามารถในการใช้โปรแกรมวาดแบบด้วย เพื่อที่จะสร้างไฟล์ dxf สำหรับวาดภาพ และในบางส่วนของการคำนวณในโปรแกรมก็ทำให้เกิดค่าความผิดพลาดของเครื่องจักรได้ เช่น การปัดเศษของการคำนวณบางอย่าง เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.3 ข้อเสนอแนะ

ในส่วนของซอฟท์แวร์นั้นในการรับไฟล์จากโปรแกรมวาดแบบควรจะรับข้อมูลไฟล์ที่เป็นลักษณะของภาพ 3 มิติ แล้วทำการสร้างคำสั่งควบคุมเครื่องจักร เพราะในการกำหนดความลึกของงานจากรูปด้านบนนั้นยังไม่ใช่การทำงานในลักษณะของ 3 มิติที่สมบูรณ์ เพราะจะยังไม่สามารถกำหนดพื้นผิวโค้ง พื้นเอียง หรือ พื้นผิวแบบอื่นๆ ได้ ดังนั้น ควรพัฒนาในส่วนรับข้อมูลจากโปรแกรมวาดแบบให้รับภาพที่เป็นพื้นผิว 3 มิติให้ได้ เพื่อให้สร้างชิ้นงานแบบใดๆก็ได้ และควรมีส่วนที่ใช้สร้างภาพแบบง่ายๆอย่าในโปรแกรมด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

กวิน สนธิเพิ่มพูน, รายงานการวิจัยฉบับสมบูรณ์ เรื่อง การสร้างส่วนควบคุมเครื่องกัดแนวดิ่งซีเอ็นซี ระยะที่ 1 , ศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ, 2534.

ชาลี ตระการกูล, เทคโนโลยีซีเอ็นซี, สมาคมส่งเสริมเทคโนโลยี(ไทย-ญี่ปุ่น),2538

รวิ อุดตมธนินทร์ กวิน สนธิเพิ่มพูน และไพรัช ธัชยพงษ์ ,"การพัฒนาสร้างต้นแบบส่วนควบคุมเครื่องกัดแนวดิ่งซีเอ็นซี" ,การประชุมวิชาการวิศวกรรมไฟฟ้าครั้งที่ 17 ,กรุงเทพ : สถาบันเทคโนโลยีพระจอมเกล้าพระนครเหนือ

D.Hearn&M.Baker,Computer Graphics,Prentice-Hall,1986

FANUC LTD , FANUC OM-MODEL A operator's manual,FANUC LTD,1985.

Larry Horath, Computer Numerical Control Programming of Machines,1993.

S. Harrington,Computer Graphics,A Programming Approach,McGraw-Hill,1987

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
#include<alloc.h>
#include<ctype.h>
#include<io.h>
#include<dos.h>
#include<string.h>
#include<bios.h>
#include<math.h>
#include<conio.h>
#include<dir.h>
#include<stdio.h>
#include<graphics.h>
#include<stdlib.h>
#include<fcntl.h>
#include "d:\tone\lang\tc30\bin\1\kbdcode.h"

#define COLFILE WHITE
#define SCREEN800X 10

#define FEEDMAX 4000

#define DO_CYCLE 1
#define DO_SIM   2

#define AXIS_X 1
#define AXIS_Y 2
#define AXIS_Z 3

#define HCTL_X 0x300
#define HCTL_Y 0x303
#define HCTL_Z 0x306

#define FACINCH 183
#define FACCENT 464

#define REG_POS 0x2f
#define REG_VEL 0x27

#define CW_DIR  1
#define CCW_DIR 2

#define KEY_LEFT     75
#define KEY_RIGHT    77
#define KEY_UP       72
#define KEY_DOWN     80
#define KEY_PGUP     73
#define KEY_PGDW     81
#define KEY_HOME     71
#define KEY_END      79
#define ESC          27
#define ENTER        13

#define MOVE_X       4
#define MOVE_Y       2
```

เอกสารนี้มีลิขสิทธิ์ สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
#define MOVE_Z      1
#define MOVE_XY     6
#define MOVE_XZ     5
#define MOVE_YZ     3

#define BUFMAX                  96
#define COL_CUT_VER             CYAN
#define COL_CUT_HOR             BROWN
#define COL_CUT_NORM            YELLOW
#define COL_OBJECT              YELLOW
#define COL_DXF3D_OBJ           LIGHTRED
#define COL_DXF3D_LINE          LIGHTRED
/*---------- sim cut ----------*/
#define HOMEX       150
#define HOMEY       150
#define HOMEZ       150
#define FACMP       2
#define SCALEOBJ    150.0
#define COLRED      64
#define COLGREEN    128
#define COLBLUE     192
#define COLORPATH   WHITE

#define CODETEMP     "code.tmp"
#define CODENAME     "mch.tmp"
#define OBJECTNAME    "temp.obj"

// #define path_data_file "d:\\tone\\lang\\data\\"
#define path_data_file  "data\\"
#define cuprintpath     "data\\"
#define FNAME           "data\\n1.dxf"
#define OFFSETNAME  "data\\offset.tmp"
#define CUTNAME     "data\\cut.tmp"

#define MCHNAME      "data\\mch.cod"
#define ISONAME     "data\\mch.iso"

#define ON          1
#define OFF         0
#define STEPMOVE    0.8

#define NONENTITY   0
#define POINT       1
#define LINE        2
#define FACE        3
#define SECTION     4
#define ENDSEC      5
#define ENDTAB      6
#define TABLE       7
#define CIRCLE      8
#define ARC         9
#define POLYLINE    10
#define VERTEX      11
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
#define SEQEND      12
#define VERT_B      13
/*-------------------------------------------------------*/
#define linestep    5

#define TOOLMAX     15
#define YADJ        20

#define INCH         1
#define CENT         2

#define MOVE_ABS    1
#define MOVE_INC     2

#define SHOWDXF     1
#define SHOWFILL    2
#define SHOWTEST    3
#define SHOWCUT     4

#define THAIENG     1
#define NUMBER      2
#define ONEORTWO    3
#define ENGLISH     4
#define ENGUPCASE   5

#define MAXMAINMENU 7
#define OFF         0
#define ON          1
#define NO          0
#define YES         1

#define STEPMOVE    0.8

#define BYTE        15
#define Y           20

#define MENU_H      32

/*---------------------- d i r  var ----------------------*/
#define DIRCOLMAX   5
#define DIRROWMAX   5
/*---------------------- d x f ----------------------------*/
#define POLYMAX     15

#define ON          1
#define OFF         0
#define IMAX        15

/*-------------------------------------------------------*/
int FlagAction;
void MoveMachine(double x1,double y1,double z1,double x2,double y2,double z2,double feed);
void MoveMachineTo(double mx,double my,double mz,double feed);
void set_move(int move);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void set_vel(int axis,double vl);
void set_pos(int axis,double ps);
double get_pos(int axis);
/*-------------------------------------------------------*/

struct point{
  double x,y;
};

struct pic_offset{
  double x,y,r;
  char s;
};

struct pic{
  double x[IMAX],y[IMAX],r[IMAX];
  int s[IMAX];
};

struct frame{
  char name[60];
  int num;  /*-- number of picture in TOP --*/
  int n[POLYMAX];
  int type[POLYMAX];
  double xmax[POLYMAX],ymax[POLYMAX];
  double xmin[POLYMAX],ymin[POLYMAX];
  struct pic poly[POLYMAX];
};

typedef struct {
                        int l,t,r,b,max;
                        char *text,*save,*help;
} PULLDOWN; /* STRUCTURE OF PULLDOWN MENU */

struct dirnode {
      char filename[13];
      struct dirnode *previous;
      struct dirnode *next;
};

/*-------------------------------------------------------*/
void draw_head(void);
void get_value_feedrate(void);
int show_project_onscreen(void);
void save_project(void);
void load_project(void);
void load_dxf(void);
void load_machine(void);
void set_feed_rate(void);
void set_spindle_rate(void);
void set_jog_speed(void);
void jog_mode(void);
void out_position(void);
```

เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
/*---------------------- S I M U L A T I O N ----------*/
void InitBuffer(void);
void simulate(void);
double readnum(char text[],int *i);
void do_gcode_00(int doFlag,double x,double y,double z,int moveFlag);
void do_gcode_01(int doFlag,double x,double y,double z,int moveFlag,double feedrate);
void do_gcode_02(int doFlag,double numX,double numY,double numZ,double numI,double numJ,double numK,int moveFlag,double feedrate);
void do_gcode_03(int doFlag,double numX,double numY,double numZ,double numI,double numJ,double numK,int moveFlag,double feedrate);
void do_gcode_04(double numP);
void do_gcode_20(void);
void do_gcode_21(void);
void do_gcode_90(void);
void do_gcode_91(void);

void do_mcode_02(void);
void do_mcode_03(void);
void do_mcode_04(void);
void do_mcode_05(void);

void toggle(void);
void real(void);
void Menubar(int s);
/*---------------------- o b j e c t ----------------*/
void init_dxf_data(void);
unsigned int mm2point(unsigned int mm);
unsigned int point2mm(unsigned int pt);
unsigned int InitObject(unsigned int x,unsigned int y,unsigned int z);
unsigned int ReadXObject(void);
unsigned int ReadYObject(void);
unsigned int ReadZObject(void);
unsigned int ReadObject(int x,int y);
unsigned int WriteObject(int x,int y,unsigned int z);
void CloseObject(void);
void Moveline(double x1,double y1,double z1,double x2,double y2,double z2);;
void MoveLineTo(double x,double y,double z);
void MoveLineInc(double x,double y,double z);
void HomeX(void);
void HomeY(void);
void HomeZ(void);
void MoveArcXY(int dir,double cx,double cy,double r,double p,double st,double end);
void MoveArcYZ(int dir,double cy,double cz,double r,double p,double st,double end);
void MoveArcXZ(int dir,double cx,double cz,double r,double p,double st,double end);
void setColor256(void);
void showColor256(void);
void TestObject(void);
void ViewSolidObject(void);
void ViewSolidObject64(void);
/*---------- Init Graphics ----------------------*/
void initialize_graph(void);
void init_system(void);
void getvalue_tools(void);
/*---------------------- Show dxf ----------------------*/
void compile_code(void);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void getvalue_deep(void);
int read_file_dxf(void);
int show_view_dxf_on_screen(void);
void show_scale_dxf_onscreen(void);
void draw_deep_dxf(void);
void calc_scale_dxf(void);
void Circledxf(double x,double y,double r);
void Linedxf(double x1,double y1,double x2,double y2);
void Rectangledxf(double x1,double y1,double x2,double y2);
void draw_dxf_3d(void);
void draw_dxf_fill(int mode);
/*---------- C N C function ----------*/
void save_machine_code(void);
/*---------- 3 d ----------*/
void init3d(double x,double y,double z);
void init3d_all_view(void);
void line3df(double x1,double y1,double z1,double x2,double y2,double z2,int col);
void plot3df(double x,double y,double z,int col);
void outtext3d(double x,double y,double z,char *text,int col);
void line3d(double x1,double y1,double z1,double x2,double y2,double z2,int col);
void arc3dXY(int,double,double,double,double,double,double,int);
void arc3dXZ(int,double,double,double,double,double,double,int);
void arc3dYZ(int,double,double,double,double,double,double,int);
void axis(double ln);
void block3d(double,double,double,double,double,double,int);
void block3dhide(double,double,double,double,double,double,int);
void circle_z3d(double x,double y,double z,double r,int col);
void dispblank(int x,int y,int maxlen,int attr);
/*---------- d x f  f i l e ----------*/
void getvalue(int i,char *text,double *v);
int open_dxf_file(char *fname);
int search_entities(void);
void trim00(char *text[]);
void read_dxf_data(void);
void readxyz_dxf(double *x,double *y,double *z,FILE *data);
void pointto(double x,double y);
void add_circle(double x,double y,double r);
void init_poly(double x,double y);
void add_poly(double x,double y,double r);
void draw_dxf_data(void);
void calc_dxf_max_data(void);
void sort_data_polyline(void);
void draw_fill(void);
/*---------- F O N T  F U N C . ----------*/
void widthfont(void);
void loadfont(void);
void initthaifont(void);
void drawfont(int x,int y,int c,int col);
void drawfontbackground(int x,int y,int c,int col,int bak1,int bak2);
void drawtext(int x,int y,char text[],int col);
void dispblank_xy_text(int x,int y,int maxlen,int col);
void drawtextbackground(int x,int y,char text[],int col,int bkg1,int bkg2);
void showfont(void);
```

เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
/*---------- S C R E E N  F U N C. ----------*/
void frame(void);
/*---------- D I R function  ----------*/
int selectfile(char mask[]);
void dirleft(void);
void dirright(void);
void dirdown(void);
void dirup(void);
void dirpgdn(void);
void setdirpos(struct dirnode *namewant);
void dirpgup(void);
void freedir(void);
void showpagedir(void);
void showfile(int col,int row,char attr);
int createdir(char mask[]);
/*-------------------------------------------*/
FILE *datadxf;
FILE *datacut;
FILE *cutfile,*mchfile,*isofile;
FILE *proj;
FILE *object,*code,*tempcode;
/*-------------------------------------------*/
int move;
int fastspeed=500,slowspeed=100,curvespeed=100;
double facmove;
double feedrate_value=100.0,spindle_value=1500.0,jog_value=50.0;
double scalev4x,scalev4y,scalev4z;
double abs_pos_x,abs_pos_y,abs_pos_z;
double rel_pos_x,rel_pos_y,rel_pos_z;
double set_pos_x,set_pos_y,set_pos_z;
double homex,homey,homez;
/*-------------------------------------------*/
struct dirnode *headdir, *dirpage;
int dircol;
int dirrow;
/*-------------------------------------------*/
struct frame currentlayer;
int GRAPH=ON;
int SIMGRAPH;
/*---------- d x f ----------*/
double toolsize[TOOLMAX],objectdeep[TOOLMAX],tooldeep[TOOLMAX];
int toolno[TOOLMAX];
/*---------- 3 d i m e n s i o n ----------*/
double scale3d=1.0/2.0;
double xp,yp,zp;
int view1,view2,view3,view4,view5;
int CUT;
/*-------------------------------------------*/
int unit=INCH;
double scaleunit=254;
double objectwidth=100.0,
       objectlength=100.0,
       objectheight=100.0,
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    objectfac;
double dispobjx,dispobjy,dispobjz;
int TIME=0;
int scale=ON;
/*-------------------------------------------------*/
int readkbd(void);
int Selection(PULLDOWN *p,int *down,int *vtab);
PULLDOWN *pop_menu[MAXMAINMENU+1];
int i,j,*dn_sel,mainmenu=1,row[MAXMAINMENU+1];
char *menu[MAXMAINMENU]={  "File",
                           "ชิ้นงาน เครื่องมือ",
                           "โปรแกรมคุมเครื่องจักร",
                           "จำลองการทำงาน",
                           "ตั้งค่า",
                           "Jog mode",
                           "Cycle"}; /* title menu bar */
char dfont [5121];/*-char ditalicfont [5121];-*/
char wfont[256];
int GraphDriver;  /* The Graphics device driver  */
int GraphMode;    /* The Graphics mode value     */
int ErrorCode;    /* Reports any graphics errors */
int maxx,maxy;

double x_p,y_p,z_p;
double ex,ey,ez;
double sxv0,syv0,szv0,cxv0,cyv0,czv0;
double sxv1,syv1,szv1,cxv1,cyv1,czv1;
double sxv2,syv2,szv2,cxv2,cyv2,czv2;
double sxv3,syv3,szv3,cxv3,cyv3,czv3;
double sxv4,syv4,szv4,cxv4,cyv4,czv4;
double sxv5,syv5,szv5,cxv5,cyv5,czv5;
double degx,degy,degz;

int ch;
unsigned char buffer[BUFMAX][BUFMAX];
int tog=1;
int quit=OFF;

/*-------------------------------------------------*/
int nobject;
double dxfx1,dxfx2,dxfy1,dxfy2;
double dxfx,dxfy;
double scaledxfx,scaledxfy;
double dxfpx,dxfpy;

char tmpname[60];
char prjfile[60];
char dxffile[60];
char mchname[60];
char text[255];
/*---------- S I M C U T ----------*/
int CUT=OFF;
int view4=ON;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
char tnum[10];
int nRead=255;
int labelcount=0;
long labelseek[100];
char labelname[100][10];
int labelnum[100];
unsigned char variable[100];
double toolx,tooly,toolz,toolr=2,toolh,ToolT;
double tablex,tabley,tablez;
unsigned char objectx,objecty,objectz;
double relposx,relposy;
double xp,yp,zp;
double sx,sy,sz,cx,cy,cz;
double degx,degy,degz;

int GraphDriver;  /* The Graphics device driver  */
int GraphMode;    /* The Graphics mode value     */
int ErrorCode;    /* Reports any graphics errors */

int maxx,maxy;
int ch;
int i,j;
unsigned int tu;
double ObjectX,ObjectY;
double posX,posY,posZ,posOldX,posOldY,posOldZ;

void Initialize_graph(void);
int huge detectVESA(void);
#define SEL_MENU BLACK

unsigned int mm2point(unsigned int mm)
{
  return((unsigned int)(mm/FACMP));
}

unsigned int point2mm(unsigned int pt)
{
  return((unsigned int)(pt*FACMP));
}

unsigned int InitObject(unsigned int x,unsigned int y,unsigned int z)
{
  double dmax;
  unsigned int i,j;

  dmax=x;
  if(y>dmax)dmax=y;
  if(z>dmax)dmax=z;
  objectfac=(double)(BUFMAX)/dmax;

  objectwidth  = x;
  objectlength = y;
  objectheight = z;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  objectx = (unsigned int)(x*objectfac);
  objecty = (unsigned int)(y*objectfac);
  objectz = (unsigned int)(z*objectfac);
  for(i=0;i<BUFMAX;i++)
    for(j=0;j<BUFMAX;j++)
      buffer[i][j]=0;

  for(i=0;i<objectx;i++)
    for(j=0;j<objecty;j++)
      buffer[i][j]=objectz;

  return(0);
}

unsigned int ReadXObject(void)
{
  return((unsigned int)(objectwidth));
}

unsigned int ReadYObject(void)
{
  return((unsigned int)(objectlength));
}

unsigned int ReadZObject(void)
{
  return((unsigned int)(objectheight));
}

unsigned int ReadObject(int x,int y)
{
  unsigned char xx,yy;
  double zz;

  if(x<0) return(0xff);
  if(y<0) return(0xff);
  xx=x*objectfac;
  yy=y*objectfac;

  if( ((xx<BUFMAX)) && ((yy<BUFMAX)) )
    zz=buffer[xx][yy]/objectfac;
  else
    zz=0xff;

  return( (unsigned int)(zz) );
}

unsigned int WriteObject(int x,int y,unsigned int z)
{
  unsigned int xx,yy,zz;

  if(x<0) return(0xff);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{
  if(y<0) return(0xff);

  xx=x*objectfac;
  yy=y*objectfac;
  zz=z*objectfac;

  if( ((xx<BUFMAX)) && ((yy<BUFMAX)) )
    buffer[xx][yy]=zz;
  else
    return(0xff);
  return(0);
}

void CloseObject(void)
{
  int i,j;

  for(i=0;i<BUFMAX;i++)
    for(j=0;j<BUFMAX;j++)
      buffer[i][j]=0;
}


void real(void)
{
}

void toggle(void)
{
  if(unit==INCH)
    unit=CENT;
  else
    unit=INCH;
}

/*------------------ menu ------------------------*/

void Pull(PULLDOWN *p)
{

  setfillstyle(SOLID_FILL,GREEN);
  bar(p->l+1,p->t+1,p->r-1,p->b-1);
  setcolor(WHITE);
  outtextxy(p->l+4,p->t+4,p->text);
}



void Restor(PULLDOWN *p)
{
  setfillstyle(SOLID_FILL,BLACK);
  bar(p->l,p->t,p->r,p->b);
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void Rewrite(PULLDOWN *p,int row)
{
  int c,b,i;
  char txt[60];
  int col;
  int hight=30; /*-18;-*/

  setfillstyle(SOLID_FILL,LIGHTRED);
  bar(p->l,p->t,p->r,p->b);
  b=0;i=0;
  for(c=0;c<p->max;c++)
  {
    i=0;
    while(p->text[b]!='\n')
    {
      txt[i]=p->text[b];
      i=i+1;b=b+1;
    }
    txt[i]=0;b=b+1;
    if(row-c==1)
    {
      setfillstyle(SOLID_FILL,LIGHTGRAY);
      setcolor(SEL_MENU);
      col=SEL_MENU;
    }
    else
    {
      setfillstyle(SOLID_FILL,LIGHTBLUE);
      setcolor(WHITE);
      col=WHITE;
    }
    bar(p->l+4,p->t+(c)*hight+5,p->r-4,p->t+(c)*hight+hight);
    drawtext(p->l,p->t+(c)*hight+8,txt,col);
  }
}


void Selmenubar(int s)
{
  int l;
  int hight=31;
  int col;
  int x1,x2;

  setfillstyle(SOLID_FILL,LIGHTRED);
  bar(0,0,maxx,hight+2);

  setcolor(WHITE);
  s=s-1;
  for (i=0;i<MAXMAINMENU;i++)
  {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    x1=pop_menu[i+1]->l;
    if(il=MAXMAINMENU-1)
    {
      if(il=MAXMAINMENU-2)
              x2=pop_menu[i+2]->l;
      else
              x2=580;
    }
    else
    {
      x1=580;
      x2=maxx-2;
    }
    if(i==s)
    {
      setfillstyle(SOLID_FILL,LIGHTGRAY);
      bar(x1+3,2,x2-2,hight);
      setcolor(SEL_MENU);
      col=SEL_MENU;
    }
    else
    {
      setfillstyle(SOLID_FILL,LIGHTBLUE);

      bar(x1+3,2,x2-2,hight);
      setcolor(WHITE);
      col=WHITE;
    }
    drawtext(x1+10,5,menu[i],col);
  }
}


void Menubar(int s)
{
  int i;
  int col;
  int x1,x2;

  int hight=31; /*-17;-*/


  setfillstyle(SOLID_FILL,LIGHTRED);
  bar(0,0,maxx,hight+2);
  setcolor(WHITE);
  s=s-1;
  for (i=0;i<MAXMAINMENU;i++)
  {
    x1=pop_menu[i+1]->l;
    if(il=MAXMAINMENU-1)
    {
      if(il=MAXMAINMENU-2)
              x2=pop_menu[i+2]->l;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    else
            x2=580;
  }
  else
  {
   x1=580;
   x2=maxx-2;
  }

  if(i==s)
  {
   setfillstyle(SOLID_FILL,LIGHTGRAY);
   bar(x1+3,2,x2-2,hight);

   setcolor(SEL_MENU);
   col=SEL_MENU;
  }
  else
  {
   setfillstyle(SOLID_FILL,LIGHTBLUE);
   bar(x1+3,2,x2-2,hight);

   setcolor(WHITE);
   col=WHITE;
  }   drawtext(x1+10,5,menu[i],col);
 }
}

int Selection(PULLDOWN *p,int *down,int *vtab)
{
 int se,mainmenu=*down,row;
 int old=mainmenu;

 row=*vtab;
 do
 {
  Selmenubar(mainmenu);
  Rewrite(p,row);
  se=getch();
  if (se==0)
  {
   se=getch();
   switch(se)
   {
        case 72: row--;
                 if(row<1) row=p->max;
                 break;
        case 80: row++;
                 if(row>p->max) row=1;
                 break;
        case 75: mainmenu--;
                 if(mainmenu<1) mainmenu=MAXMAINMENU;
                 break;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            case 77: mainmenu++;
                     if(mainmenu>MAXMAINMENU) mainmenu=1;
                     break;
            default: break;
        }
      }
    } while(sel=13&&old==mainmenu);
    Restor(p);
    *vtab=row;
    if (se==13)
    {
      *down=row;
      return(mainmenu);
    }
    else
    {
      *down=mainmenu;
      return(0);
    }
}

void About(void)
{
  PULLDOWN *p;

  p=(PULLDOWN *)malloc(sizeof(PULLDOWN));
  p->l=200;
  p->t=200;
  p->r=600;
  p->b=400;
  p->text="CNC VERTICAL MILLING MACHINE VER 1.01";

  p->save=NULL;
  Pull(p);
  getch();
  cleardevice();
/*- rew  Restor(p); -*/
}


int huge detectVESA(void)
{  return(2); }

/* Initialize the graphics system and report any errors occur. */
void Initialize_graph(void)
{
  GraphDriver = installuserdriver("bgi256v2",detectVESA);
  if (grOk != graphresult() )
  {
    printf("Error installing VESA driver");
    exit(1);
  }
  GraphDriver = DETECT;
```

เอกสารนี้มีลิขสิทธิ์ สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  initgraph( &GraphDriver, &GraphMode, "" );
  ErrorCode = graphresult();
  if (ErrorCode != grOk)            /* Check if error occured */
  {
    printf(" Graphics System Error: %s\n",grapherrormsg(ErrorCode));
    exit(1);
  }
  maxx=getmaxx();
  maxy=getmaxy();
}

void setcolor256(void)
{ int i;

  for(i=17;i<64;i++)    setrgbpalette(i,i,i,i);

  for(i=0;i<64;i++)     setrgbpalette(i+ 64, i,50,50);
  for(i=0;i<64;i++)     setrgbpalette(i+128, 50,i,50);
  for(i=0;i<64;i++)     setrgbpalette(i+192, 50,50,i);

  for(i=0;i<64;i++)     setrgbpalette(i+ 64, i,0,0);
  for(i=0;i<64;i++)     setrgbpalette(i+128, 0,i,0);
  for(i=0;i<64;i++)     setrgbpalette(i+192, 0,0,i);
}

/*------------------ position display ----------------*/

void gph_display(void) /* print pattern of position displaying */
{
   outtextxy(20,450+ 60,"ACTUAL POSITION (ABSOLUTE)");
   outtextxy(20,450+ 75,"  X:");
   outtextxy(20,450+ 90,"  Y:");
   outtextxy(20,450+105,"  Z:");

   outtextxy(300,450+ 60,"ACTUAL POSITION (RELATIVE)");
   outtextxy(300,450+ 75,"  X:");
   outtextxy(300,450+ 90,"  Y:");
   outtextxy(300,450+105,"  Z:");

   real();

   if (i<y==1)
   {
     outtextxy(200,450+105," mm./min");
   }
     else
   {
     outtextxy(200,450+105," in./min");
   }
 }

 /*------------------ F O N T function ------------------*/
 void widthfont(void)
```

เอกสารนี้เป็นการที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{
  int i;

  for(i=0;i<255;i++)
  {
    wfont[i]=8;
  }
  wfont[209]=-8;
  wfont[212]=-8;
  wfont[213]=-8;
  wfont[214]=-8;
  wfont[215]=-8;
  wfont[216]=-8;
  wfont[217]=-8;
  wfont[218]=-8;
  wfont[219]=-8;
  wfont[220]=-8;

  wfont[231]=-8;
  wfont[232]=-8;
  wfont[233]=-8;
  wfont[234]=-8;
  wfont[235]=-8;
  wfont[236]=-8;
  wfont[237]=-8;
  wfont[238]=-8;
}

void loadfont(void)
{
  int handle;
  char s1[55];

  strcpy(s1,cuprintpath);
  strcat(s1,"normal.fon");
  /*-  strcat(s1,"italic.FON"); -*/
  if ((handle = open(s1,O_RDONLY || O_BINARY)) != -1) {
            read(handle,dfont,5120);
    close(handle);
  } else {
    puts("FONT FILE (NORMAL.FON) NOT FOUND !");
    exit(1);
  }

  widthfont();
}

void drawfont(int x,int y,int c,int col)
{
  int ch,i,j,a=1;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
 for(j=0;j<20;j++)
 {
  ch=dfont[c*20+j];
  for(i=0;i<8;i++)
  {
   a=0x80;
   a=a>>i;
   if((a&ch)!=0)
            putpixel(x+i,y+j,col); /*-WHITE);-*/
  }
 }
}

void drawfontbackground(int x,int y,int c,int col,int bak1,int bak2)
{
 int ch,i,j,a=1;


 for(j=0;j<20;j++)
 {
  ch=dfont[c*20+j];
  for(i=0;i<8;i++)
  {
   a=0x80;
   a=a>>i;
   if(col!=0)
   {
            if((a&ch)!=0)
              putpixel(x+i,y+j,col); /*-WHITE);-*/
            else
            {
              if(getpixel(x+i,y+j)!=col)
                putpixel(x+i,y+j,bak1); /*BLACK);*/ /*-WHITE);-*/
            }
   }else
   {
            if((a&ch)!=0)
              putpixel(x+i,y+j,RED); /*-WHITE);-*/
            else
            {
              if(getpixel(x+i,y+j)!=RED)
                putpixel(x+i,y+j,bak2); /*DARKGRAY);*/
            }
   }
  }
 }
}


 void drawtext(int x,int y,char text[],int col)
 {
  int i=0,d=0;
  int c;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  do{
    c=(unsigned char)(text[i]);
    if(wfont[c]<0)
      drawfont(x+d+wfont[c],y,c,col);
    else
    {
      drawfont(x+d,y,c,col);
      d=d+wfont[c];
    }
    i++;
  } while(text[i]!=0);
}

void drawtextbackground(int x,int y,char text[],int col,int bkg1,int bkg2)
{
  int i=0,d=0;
  int c;

  while(text[i]!=0)
  {
    c=(unsigned char)(text[i]);
    if(wfont[c]<0)
      drawfontbackground(x+d+wfont[c],y,c,col,bkg1,bkg2);
    else
    {
      drawfontbackground(x+d,y,c,col,bkg1,bkg2);
      d=d+wfont[c];
    }
    i++;
  }
}

void showfont(void)
{
  int c,i;
  int x,y;
  int col=WHITE;

  x=10;y=0;i=0;
  for(c=0;c<256;c++)
  {
    drawfont(x,y,c,col);
    x+=10;i++;
    if(i==25)
    {
      i=0;
      x=10;
      y+=25;
    }
  }
  getch();
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void initthaifont(void)
{
  loadfont();
  widthfont();
}


void dispstrhgc(char *text,int col,int row,int attr)
{
  drawtextbackground(col*8,row*22,text,attr,BLUE,DARKGRAY);/*- dir -*/
}


void errorsound(void)
{
  int i;

  for(i=500;i<800;i++)
  {
    sound(i);
    delay(3);
  }
  nosound();
}

int createdir(char mask[])
{
    struct dirnode *tempdir;
    struct ffblk ffblock;
    int done,i;
    char *source;

    headdir = (struct dirnode *) malloc(sizeof(struct dirnode));
    headdir->previous = headdir;
    headdir->next = headdir;
    tempdir = headdir;
    done = findfirst(mask,&ffblock,0);

    while (!done) {
      tempdir->next = (struct dirnode *) malloc(sizeof(struct dirnode));
      if (tempdir->next == NULL) {
              errorsound();
              /*- rew blockmsg(10); -*/
        dispstrhgc("หน่วยความจำไม่พอ ! กด <ESC> เพื่อทำงานต่อ",22,10,2);
        while (ebioskey(0) != ESCKEY);
        freedir();
        return(NO);
      }
      source = ffblock.ff_name;
      i = 0;
      while (*source != '\0') {
        (tempdir->next)->filename[i] = *(source++);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    '   ';
      i++;
    }
    (tempdir->next)->filename[i] = '\0';
    (tempdir->next)->previous = tempdir;
    tempdir = tempdir->next;
    tempdir->next = headdir;
    headdir->previous = tempdir;
    done = findnext(&ffblock);
  }
  return(YES);
}

void showfile(int col,int row,char attr)
{
  struct dirnode *tempdir;
  int count;

  tempdir = dirpage;
  for (count = (DIRCOLMAX * row) + col;(count != 0) && (tempdir != headdir);count--)
    tempdir = tempdir->next;
  if (tempdir != headdir)
    dispstrhgc(tempdir->filename,col*13 + 9 +SCREEN800X,row+9,attr);
}

void showpagedir(void)
{
  int col,row;
  /*
  dispstrhgc(" ¯••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••™ ",9-CENTER_FACTOR,8,2);
  dispstrhgc(" –                                                                    – ",9-CENTER_FACTOR,9,2);
  dispstrhgc(" –                                                                    – ",9-CENTER_FACTOR,10,2);
  dispstrhgc(" –                                                                    – ",9-CENTER_FACTOR,11,2);
  dispstrhgc(" –                                                                    – ",9-CENTER_FACTOR,12,2);
  dispstrhgc(" –                                                                    – ",9-CENTER_FACTOR,13,2);
  dispstrhgc(" ¨••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••› ",9-CENTER_FACTOR,14,2);
  */

  framebox(4+SCREEN800X, 8, 4+70+SCREEN800X, 14, BLUE);

  for (row = 0;row != DIRROWMAX;row++)
    for (col = 0;col != DIRCOLMAX;col++)
        showfile(col,row,COLFILE);
}

void freedir(void)
{
  struct dirnode *tempdir,*tempdir2;
  tempdir = headdir->next;
  while (tempdir != headdir) {
    tempdir2 = tempdir;
    tempdir = tempdir->next;
    free(tempdir2);
  }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
}

void dirpgup(void)
{
    struct dirnode *tempdir;
    int count,countrow;
    tempdir = dirpage;
    for (countrow = DIRROWMAX;(countrow != 0) && (tempdir->previous != headdir);countrow--)
      for (count = DIRCOLMAX;(count != 0) && (tempdir->previous != headdir);count--)
        tempdir = tempdir->previous;
    if (tempdir->previous != headdir)
      dirpage = tempdir;
    else {
      dirpage = headdir->next;
      dirrow = 0;
    }
    showpagedir();
}

void setdirpos(struct dirnode *namewant)
{
    int count;
    struct dirnode *tempdir;
    tempdir = dirpage;
    dirrow = 0;
    dircol = 0;
    while(tempdir != namewant) {
      for (count = (DIRCOLMAX - 1);(count != 0) && (tempdir != namewant);count--) {
        tempdir = tempdir->next;
        dircol++;
      }
      if (tempdir != namewant){
        dirrow++;
        dircol = 0;
        tempdir = tempdir->next;
      }
    }
}

void dirpgdn(void)
{
   struct dirnode *tempdir;
   int count,countrow;
   tempdir = dirpage;
   for (countrow = DIRROWMAX;(countrow != 0) && (tempdir->next != headdir);countrow--){
     for (count = DIRCOLMAX;(count != 0) && (tempdir->next != headdir);count--)
         tempdir = tempdir->next;
     if (tempdir->next != headdir)
         dirpage = tempdir;
   }
   for (count = dircol;(count != 0) && (tempdir->next != headdir);count--)
      tempdir = tempdir->next;
   showpagedir();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  ''
  setdirpos(tempdir);
}

void dirup(void)
{
  struct dirnode *tempdir;
  int count;
  tempdir = dirpage;
  for (count = DIRCOLMAX;(count != 0) && (tempdir->previous != headdir);count--)
      tempdir = tempdir->previous;
  if (count == 0){
    dirpage = tempdir;
    showpagedir();
  }
}

void dirdown(void)
{
  struct dirnode *tempdir,*tempdir2;
  int count;
  tempdir = dirpage;
  for (count = (DIRCOLMAX * dirrow);(count != 0) && (tempdir->next != headdir);count--)
     tempdir = tempdir->next;
  tempdir2 = dirpage;
  for (count = (DIRCOLMAX - 1);(count != 0) && (tempdir->next != headdir);count--){
     tempdir = tempdir->next;
     tempdir2 = tempdir2->next;
  }
  if (tempdir->next != headdir){
    tempdir2 = tempdir2->next;
    if (dirrow == (DIRROWMAX - 1)){
            dirpage = tempdir2;
      showpagedir();
    }
    for (count = (dircol + 1);(count != 0) && (tempdir->next != headdir);count--)
       tempdir = tempdir->next;
    setdirpos(tempdir);
  }
}

void dirright(void)
{
  struct dirnode *tempdir;
  int count;
  tempdir = dirpage;
  for (count = (DIRCOLMAX * dirrow) + dircol;count != 0;count--)
     tempdir = tempdir->next;
  if (tempdir->next != headdir)
    dircol++;
}

void dirleft(void)
{
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  struct dirnode *tempdir;
  int count;
  tempdir = dirpage;
  for (count = (DIRCOLMAX * dirrow) + DIRCOLMAX - 1;(count != 0) && (tempdir->next != headdir);count--)
     tempdir = tempdir->next;
  setdirpos(tempdir);
}

int selectfile(char mask[])
{
    struct dirnode *tempdir;
    int count,c,i,j;

    if (createdir(mask) == NO)
      return(NO);
    if (headdir->next != headdir){
      dirpage = headdir->next;
      dircol = 0;
      dirrow = 0;
      showpagedir();
      showfile(dircol,dirrow,0);
      do{
        c = ebioskey(0);
        switch(c) {
        case UPKEY : showfile(dircol,dirrow,2);
                     if (dirrow == 0)
                       dirup();
                     else
                       dirrow--;
                     showfile(dircol,dirrow,0);
                     break;
        case DNKEY : showfile(dircol,dirrow,2);
                     dirdown();
                     showfile(dircol,dirrow,0);
                     break;
        case RIKEY : showfile(dircol,dirrow,2);
                     if (dircol == (DIRCOLMAX-1))
                       dircol = 0;
                     else
                       dirright();
                     showfile(dircol,dirrow,0);
                              break;
        case LEKEY : showfile(dircol,dirrow,2);
                     if (dircol == 0)
                       dirleft();
                     else
                                    dircol--;
                     showfile(dircol,dirrow,0);
                     break;
        case PGUPKEY : dirpgup();
                       showfile(dircol,dirrow,0);
                       break;
        case PGDNKEY : dirpgdn();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                showfile(dircol,dirrow,0);
                break;
      case ESCKEY : freedir();
                mask[0] = '\0';
                return(NO);
      };
    } while (c != RETKEY);
    tempdir = dirpage;
    for (count = (DIRCOLMAX * dirrow) + dircol;(count != 0) && (tempdir->next != headdir);count--)
      tempdir = tempdir->next;
    j = strlen(mask);
    while ((j != 0) && (mask[j] != '\\') && (mask[j] != ':'))
      j--;
    if (j != 0) j++;
    for (i = 0;i < 13;i++)
      mask[j++] = tempdir->filename[i];
    freedir();
    return(YES);
  } else {
    mask[0] = '\0';
    return(NO);
  }
}

void init3d(double x,double y,double z)
{
  double sx,sy,sz,cx,cy,cz;

  sx=sin(x*M_PI/180.0);  cx=cos(x*M_PI/180.0);
  sy=sin(y*M_PI/180.0);  cy=cos(y*M_PI/180.0);
  sz=sin(z*M_PI/180.0);  cz=cos(z*M_PI/180.0);

  sxv0=ceil(sx*1.0e15)/1.0e15;
  syv0=ceil(sy*1.0e15)/1.0e15;
  szv0=ceil(sz*1.0e15)/1.0e15;

  cxv0=ceil(cx*1.0e15)/1.0e15;
  cyv0=ceil(cy*1.0e15)/1.0e15;
  czv0=ceil(cz*1.0e15)/1.0e15;
}


void init3d_all_view(void)
{

  /*-------- view 1 --------*/
  init3d(degx,degy,degz);
  cxv1=cxv0; cyv1=cyv0; czv1=czv0;
  sxv1=sxv0; syv1=syv0; szv1=szv0;

  /*-------- view 2 --------*/
  init3d(40.0,-270.0,-50.0);
  cxv2=cxv0; cyv2=cyv0; czv2=czv0;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  sxv2=sxv0; syv2=syv0; szv2=szv0;

  /*--------- view 3 ----------*/
  init3d(90.0,-180.0,0.0);
  cxv3=cxv0; cyv3=cyv0; czv3=czv0;
  sxv3=sxv0; syv3=syv0; szv3=szv0;

  /*--------- view 4 ----------*/
  init3d(-180.0,-180.0,90.0);
  cxv4=cxv0; cyv4=cyv0; czv4=czv0;
  sxv4=sxv0; syv4=syv0; szv4=szv0;

}

void plot3df(double x,double y,double z,int col)
{
  double x2,y2,z2;
  double xx,yy,zz,xxp,yyp;
  double halfx,halfy;

  x=x*scale3d;
  y=y*scale3d;
  z=z*scale3d;

  x2=x/2;
  y2=y/2;
  z2=z/2;

  xx=(x*cyv1+(y*sxv1+z*cxv1)*syv1)*czv1-(y*cxv1-z*sxv1)*szv1;
  yy=(x*cyv1+(y*sxv1+z*cxv1)*syv1)*szv1+(y*cxv1-z*sxv1)*czv1;
  zz=-x*syv1+(y*sxv1+z*cxv1)*cyv1;

  if(zp==0)
  {
    xxp=xp;
    yyp=yp;
  } else
  {
    xxp=xp/zp;
    yyp=yp/zp;
  }
  xx=xx-zz*(xxp);
  yy=yy-zz*(yyp);

  if(view1==ON)
  {
    /*-- view 3D --*/
    halfx=maxx/2.0+maxx/4.0;
    halfy=maxy/2.0+maxy/4.0+YADJVIEW4;
    putpixel(xx+halfx,yy+halfy,col);
  }
  if(view2==ON)    /*-- view Top --*/
  {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      halfx=maxx/2.0-maxx/4.0;
      halfy=maxy/2.0-maxy/4.0;
//      init3d(40.0,-270.0,-50.0);
      xx=(x2*cyv2+(y2*sxv2+z2*cxv2)*syv2)*czv2-(y2*cxv2-z2*sxv2)*szv2;
      yy=(x2*cyv2+(y2*sxv2+z2*cxv2)*syv2)*szv2+(y2*cxv2-z2*sxv2)*czv2;
      zz=-x2*syv2+(y2*sxv2+z2*cxv2)*cyv2;

      xx=xx-zz*(xxp);
      yy=yy-zz*(yyp);
      putpixel(xx+halfx,yy+halfy,col);
   }
   if(view3==ON)   /*-- view Front --*/
   {
      halfx=maxx/2.0+maxx/4.0;
      halfy=maxy/2.0-maxy/4.0;
//      init3d(90.0,-180.0,0.0);
      xx=(x2*cyv3+(y2*sxv3+z2*cxv3)*syv3)*czv3-(y2*cxv3-z2*sxv3)*szv3;
      yy=(x2*cyv3+(y2*sxv3+z2*cxv3)*syv3)*szv3+(y2*cxv3-z2*sxv3)*czv3;
      zz=-x2*syv3+(y2*sxv3+z2*cxv3)*cyv3;

      xx=xx-zz*(xxp);
      yy=yy-zz*(yyp);
      putpixel(xx+halfx,yy+halfy,col);
   }
   if(view4==ON)    /*-- view Side --*/
   {
      halfx=maxx/2.0-maxx/4.0;
      halfy=maxy/2.0+maxy/4.0;
//      init3d(-180.0,-180.0,90.0);
      xx=(x2*cyv4+(y2*sxv4+z2*cxv4)*syv4)*czv4-(y2*cxv4-z2*sxv4)*szv4;
      yy=(x2*cyv4+(y2*sxv4+z2*cxv4)*syv4)*szv4+(y2*cxv4-z2*sxv4)*czv4;
      zz=-x2*syv4+(y2*sxv4+z2*cxv4)*cyv4;

      xx=xx-zz*(xxp);
      yy=yy-zz*(yyp);
      putpixel(xx+halfx,yy+halfy,col);
   }

   if(view5==ON)  /*------  view big screen ------*/
   {
     halfx=maxx/2.0;
     halfy=maxy/2.0+YADJVIEW4;
     putpixel(xx+halfx,yy+halfy,col);
   }
  }

  void Move3d(double x,double y,double z,int col)
  {
    unsigned int a,b,c;
    int i,j,k;

    if(SIMGRAPH==ON)
    {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    abs_pos_x=x;
    abs_pos_y=y;
    abs_pos_z=z;

    //double rel_pos_x,rel_pos_y,rel_pos_z;
    out_position();

    plot3df(x,y,z,col);
  }
  if(CUT==ON)
  {
/*  toolx=x;
    tooly=y;
    toolz=z;
*/  /*— if (x and y in object ) then cut object --*/
    z=objectheight+z;
    if(   (0<=x)&&(x<=objectwidth) && (0<=y)&&(y<=objectlength)
          && (0<=z)&&(z<=objectheight) )
    {
          a=(unsigned int)(x);
          b=(unsigned int)(y);
          c=(unsigned int)(z);
          for(i=a-toolr;i<=(a+toolr);i++)
            for(j=b-toolr;j<=(b+toolr);j++)
            {
              k=ReadObject(i,j);
              if( (c<k)&&(k!=0xff))
                k=WriteObject(i,j,c);
            }
    }
  }
  delay(TIME);
}

void outtext3d(double x,double y,double z,char *text,int col)
{
  double xx,yy,zz,xxp,yyp;
  double halfx,halfy;

  x=x*scale3d;  y=y*scale3d;  z=z*scale3d;

  xx=(x*cyv1+(y*sxv1+z*cxv1)*syv1)*czv1-(y*cxv1-z*sxv1)*szv1;
  yy=(x*cyv1+(y*sxv1+z*cxv1)*syv1)*szv1+(y*cxv1-z*sxv1)*czv1;
  zz=-x*syv1+(y*sxv1+z*cxv1)*cyv1;

  if(zp==0)
  {
    xxp=xp;
    yyp=yp;
  } else
  {
    xxp=xp/zp;
    yyp=yp/zp;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  }
  xx=xx-zz*(xxp);
  yy=yy-zz*(yyp);
  setcolor(col);

  if(view4==ON)
  {
    halfx=maxx/2+maxx/4;
    halfy=maxy/2+maxy/4+YADJVIEW4;
  } else
  {
    halfx=maxx/2;
    halfy=maxy/2;
  }

  outtextxy(xx+halfx,yy+halfy,text);
}

void line3d(double x1,double y1,double z1,double x2,double y2,double z2,int col)
{
  double dx,dy,dz;
  double dmax=0;
  double stepX,stepY,stepZ;
  double count;

  if( (x1==x2)&&(y1==y2)&&(z1==z2) )
  {
    Move3d(x1,y1,z1,col);
    return;
  }

  dx=x2-x1;
  dy=y2-y1;
  dz=z2-z1;
  if( fabs(dx) > fabs(dy) ) dmax=fabs(dx);
  else dmax=fabs(dy);
  if( fabs(dz) > dmax ) dmax = fabs(dz);

  stepX=dx/dmax;
  stepY=dy/dmax;
  stepZ=dz/dmax;

  count=0;
  do{
    Move3d(x1,y1,z1,col);
    x1+=stepX;
    y1+=stepY;
    z1+=stepZ;
    count++;
  } while(count<dmax);
  Move3d(x2,y2,z2,col);
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void line3df(double x1,double y1,double z1,double x2,double y2,double z2,int col)
{
 double x,y,z;
 double xx1,yy1,xx2,yy2,zz1,zz2;
 double xx,yy,zz,xxp,yyp;
 double halfx,halfy;

 x1=x1*scale3d;  y1=y1*scale3d;  z1=z1*scale3d;
 x2=x2*scale3d;  y2=y2*scale3d;  z2=z2*scale3d;

 x=x1;y=y1;z=z1;

 xx=(x*cyv1+(y*sxv1+z*cxv1)*syv1)*czv1-(y*cxv1-z*sxv1)*szv1;
 yy=(x*cyv1+(y*sxv1+z*cxv1)*syv1)*szv1+(y*cxv1-z*sxv1)*czv1;
 zz=-x*syv1+(y*sxv1+z*cxv1)*cyv1;

 if(zp==0)
 {
   xxp=xp;      yyp=yp;
 } else
 {
  xxp=xp/zp;    yyp=yp/zp;
 }
 xx1=xx-zz*(xxp);
 yy1=yy-zz*(yyp);

 x=x2;y=y2;z=z2;

 xx=(x*cyv1+(y*sxv1+z*cxv1)*syv1)*czv1-(y*cxv1-z*sxv1)*szv1;
 yy=(x*cyv1+(y*sxv1+z*cxv1)*syv1)*szv1+(y*cxv1-z*sxv1)*czv1;
 zz=-x*syv1+(y*sxv1+z*cxv1)*cyv1;

 if(zp==0)
 {
   xxp=xp;      yyp=yp;
 } else
 {
  xxp=xp/zp;    yyp=yp/zp;
 }
 xx2=xx-zz*(xxp);
 yy2=yy-zz*(yyp);

 setcolor(col);

 if(view1==ON)   /*--- view 3d --*/
 {
    halfx=maxx/2.0+maxx/4.0;
    halfy=maxy/2.0+maxy/4.0+YADJVIEW4;
    line(xx1+halfx,yy1+halfy,xx2+halfx,yy2+halfy);
 }
 if(view2==ON)   /*--- view 1 --*/
 {
    halfx=maxx/2.0-maxx/4.0;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    halfy=maxy/2.0-maxy/4.0;
//    init3d(40.0,-270.0,-50.0);
    x=x1/2.0;y=y1/2.0;z=z1/2.0;
    xx1=(x*cyv2+(y*sxv2+z*cxv2)*syv2)*czv2-(y*cxv2-z*sxv2)*szv2;
    yy1=(x*cyv2+(y*sxv2+z*cxv2)*syv2)*szv2+(y*cxv2-z*sxv2)*czv2;
    zz1=-x*syv2+(y*sxv2+z*cxv2)*cyv2;

    x=x2/2.0;y=y2/2.0;z=z2/2.0;
    xx2=(x*cyv2+(y*sxv2+z*cxv2)*syv2)*czv2-(y*cxv2-z*sxv2)*szv2;
    yy2=(x*cyv2+(y*sxv2+z*cxv2)*syv2)*szv2+(y*cxv2-z*sxv2)*czv2;
    zz2=-x*syv2+(y*sxv2+z*cxv2)*cyv2;

    xx1=xx1-zz1*(xxp);      yy1=yy1-zz1*(yyp);
    xx2=xx2-zz2*(xxp);      yy2=yy2-zz2*(yyp);
    line(xx1+halfx,yy1+halfy,xx2+halfx,yy2+halfy);
 }
 if(view3==ON)   /*--- view 2 --*/
 {
    halfx=maxx/2.0+maxx/4.0;
    halfy=maxy/2.0-maxy/4.0;
//    init3d(90.0,-180.0,0.0);
    x=x1/2.0;y=y1/2.0;z=z1/2.0;
    xx1=(x*cyv3+(y*sxv3+z*cxv3)*syv3)*czv3-(y*cxv3-z*sxv3)*szv3;
    yy1=(x*cyv3+(y*sxv3+z*cxv3)*syv3)*szv3+(y*cxv3-z*sxv3)*czv3;
    zz1=-x*syv3+(y*sxv3+z*cxv3)*cyv3;

    x=x2/2.0;y=y2/2.0;z=z2/2.0;
    xx2=(x*cyv3+(y*sxv3+z*cxv3)*syv3)*czv3-(y*cxv3-z*sxv3)*szv3;
    yy2=(x*cyv3+(y*sxv3+z*cxv3)*syv3)*szv3+(y*cxv3-z*sxv3)*czv3;
    zz2=-x*syv3+(y*sxv3+z*cxv3)*cyv3;

    xx1=xx1-zz1*(xxp);      yy1=yy1-zz1*(yyp);
    xx2=xx2-zz2*(xxp);      yy2=yy2-zz2*(yyp);
    line(xx1+halfx,yy1+halfy,xx2+halfx,yy2+halfy);
 }
 if(view4==ON)   /*--- view 3 --*/
 {
    halfx=maxx/2.0-maxx/4.0;
    halfy=maxy/2.0+maxy/4.0;
//    init3d(-180.0,-180.0,90.0);
    x=x1/2.0;y=y1/2.0;z=z1/2.0;
    xx1=(x*cyv4+(y*sxv4+z*cxv4)*syv4)*czv4-(y*cxv4-z*sxv4)*szv4;
    yy1=(x*cyv4+(y*sxv4+z*cxv4)*syv4)*szv4+(y*cxv4-z*sxv4)*czv4;
    zz1=-x*syv4+(y*sxv4+z*cxv4)*cyv4;

    x=x2/2.0;y=y2/2.0;z=z2/2.0;
    xx2=(x*cyv4+(y*sxv4+z*cxv4)*syv4)*czv4-(y*cxv4-z*sxv4)*szv4;
    yy2=(x*cyv4+(y*sxv4+z*cxv4)*syv4)*szv4+(y*cxv4-z*sxv4)*czv4;
    zz2=-x*syv4+(y*sxv4+z*cxv4)*cyv4;

    xx1=xx1-zz1*(xxp);      yy1=yy1-zz1*(yyp);
    xx2=xx2-zz2*(xxp);      yy2=yy2-zz2*(yyp);
    line(xx1+halfx,yy1+halfy,xx2+halfx,yy2+halfy);
```

เอกสารนี้จัดทำขึ้นเพื่อการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  }
  if(view5==ON)
  {
    halfx=maxx/2.0;
    halfy=maxy/2.0+YADJVIEW4;;
    line(xx1+halfx,yy1+halfy,xx2+halfx,yy2+halfy);
  }
}

void circle_z3d(double cx,double cy,double cz,double r,int col)
{
  double stepang,stepmove;
  double count;
  double ang;
  double x,y;
  double ln;
  double x1,y1,x2,y2;

  ang=360.0;
  ln=2*M_PI*r;

  stepmove=STEPMOVE;
  stepang = stepmove*ang/ln;
  count=ang/stepang;

  //if(startang>endang)
    stepang=-stepang;
  ang=360.0; //startang;

  x1=cx+r*sin(ang*M_PI/180.0);
  y1=cy+r*cos(ang*M_PI/180.0);


  do{
    ang=ang+stepang;
    count-=1;
    x=r*sin(ang*M_PI/180.0);
    y=r*cos(ang*M_PI/180.0);
    x2 = cx+x;
    y2 = cy+y;
    line3df(x1,y1,cz,x2,y2,cz,col);
    x1=x2;  y1=y2;
   // plot3df(cx+x,cy+y,z,col);
  } while( count>=0 );
}

void arc3dXY(dir,cx,cy,pz,r,startang,endang,col)
int dir;
double cx,cy,pz,r,startang,endang;
int col;
{
  double stepang,stepmove;
  double count;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  double ang;
  double x,y;
  double ln;

  if(dir==CW_DIR)
  {
   if(startang<=endang)
     startang+=360.0;
  } else
  {
   if(startang>=endang)
     endang+=360.0;
  }
  ang=(endang-startang);
  ln=2*M_PI*r*ang/360.0;
  if(dir==CW_DIR) stepmove=-STEPMOVE;
  else stepmove=STEPMOVE;
  stepang = stepmove*ang/ln;
  count=ang/stepang;
  ang=startang;
  do{
    x=r*cos(ang*M_PI/180.0);
    y=r*sin(ang*M_PI/180.0);
/*-    plot3df(cx+x,cy+y,z,col);  -*/
    CUT=ON;
    Move3d(cx+x,cy+y,pz,col);
    if(FlagAction==DO_CYCLE)
      MoveMachineTo(cx+x,cy+y,pz,feedrate_value);
    CUT=OFF;
    ang=ang+stepang;
    count-=1;
  } while( count>=0 );
}

void arc3dXZ(dir,cx,py,cz,r,startang,endang,col)
int dir;
double cx,py,cz,r,startang,endang;
int col;
{
  double stepang,stepmove;
  double count;
  double ang;
  double x,z;
  double ln;

  if(dir==CW_DIR)
  {
   if(startang<=endang)
     startang+=360.0;
  } else
  {
   if(startang>=endang)
     endang+=360.0;
```

เอกสารนี้มีลิขสิทธิ์ที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  }
  ang=endang-startang;
  ln=2*M_PI*r*ang/360.0;
  if(dir==CW_DIR) stepmove=-STEPMOVE;
  else stepmove=STEPMOVE;
  stepang = stepmove*ang/ln;
  count=ang/stepang;
  ang=startang;
  do{
    x=r*sin(ang*M_PI/180.0);
    z=r*cos(ang*M_PI/180.0);
/*-   plot3df(cx+x,y,cz+z,col);  -*/
    CUT=ON;
    Move3d(cx+x,py,cz+z,col);
    if(FlagAction==DO_CYCLE)
      MoveMachineTo(cx+x,py,cz+z,feedrate_value);
    CUT=OFF;
    ang=ang+stepang;
    count-=1;
  } while( count>=0 );
}

void arc3dYZ(dir,px,cy,cz,r,startang,endang,col)
int dir;
double px,cy,cz,r,startang,endang;
int col;
{
  double stepang,stepmove;
  double count;
  double ang;
  double y,z;
  double ln;

  if(dir==CW_DIR)
  {
    if(startang<=endang)
      startang+=360.0;
  } else
  {
    if(startang>=endang)
      endang+=360.0;
  }
  ang=endang-startang;
  ln=2*M_PI*r*ang/360.0;
  if(dir==CW_DIR) stepmove=-STEPMOVE;
  else stepmove=STEPMOVE;
  stepang = stepmove*ang/ln;
  count=ang/stepang;
  ang=startang;
  do{
    y=r*sin(ang*M_PI/180.0);
    z=r*cos(ang*M_PI/180.0);
/*-  plot3df(x,cy+y,cz+z,col); -*/
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    CUT=ON;
    Move3d(px,cy+y,cz+z,col);
    if(FlagAction==DO_CYCLE)
      MoveMachineTo(px,cy+y,cz+z,feedrate_value);
    CUT=OFF;
    ang=ang+stepang;
    count-=1;
  } while( count>=0 );
}


void axis(double ln)
{

  line3df  (0.0,0.0,0.0,  +ln ,0.0 ,0.0,LIGHTGREEN);
  line3df  (0.0,0.0,0.0,  0.0 ,+ ln,0.0,LIGHTRED);
  line3df  (0.0,0.0,0.0,  0.0 ,0.0 ,+ln,LIGHTBLUE);

  outtext3d(0.0 + ln ,0.0    ,0.0    ,"x",LIGHTGREEN);
  outtext3d(0.0      ,0.0+ln ,0.0    ,"y",LIGHTRED);
  outtext3d(0.0      ,0.0    ,0.0+ln ,"z",LIGHTBLUE);

  if(scale==ON)
  {
    for(i=0;i<ln;i+=10)
    line3df(i,-2.0,-2.0,  i,2.0,2.0,LIGHTGREEN);
    for(i=0;i<ln;i+=10)
      line3df(-2.0,i,-2.0,  2.0,i,2.0,LIGHTRED);
    for(i=0;i<ln;i+=10)
      line3df(-2.0,-2.0,i,  2.0,2.0,i,LIGHTBLUE);
  }
}

void block3d(x1,y1,z1,x2,y2,z2,col)
double x1,y1,z1,x2,y2,z2;
int col;
{
  line3df(x1,y1,z1 , x2,y1,z1,col);
  line3df(x1,y1,z1 , x1,y2,z1,col);
  line3df(x1,y1,z1 , x1,y1,z2,col);

  line3df(x2,y2,z2 , x1,y2,z2,col);
  line3df(x2,y2,z2 , x2,y1,z2,col);
  line3df(x2,y2,z2 , x2,y2,z1,col);

  line3df(x1,y2,z2 , x1,y2,z1,col);
  line3df(x2,y2,z1 , x1,y2,z1,col);

  line3df(x1,y1,z2 , x1,y2,z2,col);
  line3df(x1,y1,z2 , x2,y1,z2,col);

  line3df(x2,y1,z1 , x2,y1,z2,col);
  line3df(x2,y1,z1 , x2,y2,z1,col);
```

เอกสารนี้จัดทำขึ้นเพื่อไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
}

void block3dhide(x1,y1,z1,x2,y2,z2,col)
double x1,y1,z1,x2,y2,z2;
int col;
{
  line3df(x1,y1,z1 , x2,y1,z1,col);
  line3df(x1,y1,z1 , x1,y2,z1,col);

  line3df(x2,y2,z2 , x1,y2,z2,col);
  line3df(x2,y2,z2 , x2,y1,z2,col);
  line3df(x2,y2,z2 , x2,y2,z1,col);

  line3df(x1,y2,z2 , x1,y2,z1,col);
  line3df(x2,y2,z1 , x1,y2,z1,col);

  line3df(x2,y1,z1 , x2,y1,z2,col);
  line3df(x2,y1,z1 , x2,y2,z1,col);
}


void framebox(int x1,int y1,int x2,int y2,int col)
{
  int Scol;

  setfillstyle(SOLID_FILL,col);
  bar(x1*8,y1*22,x2*8,y2*22);
  if(coll=DARKGRAY)
    setfillstyle(SOLID_FILL,DARKGRAY);
  else
    setfillstyle(SOLID_FILL,LIGHTGRAY);
  bar(x1*8+2,y1*22+2,x2*8-2,y2*22-2);
  setfillstyle(SOLID_FILL,col);
  bar(x1*8+4,y1*22+4,x2*8-4,y2*22-4);
  Scol=getcolor();
  setcolor(WHITE);
  rectangle(x1*8,y1*22,x2*8,y2*22);
  setcolor(Scol);
}

/*-------------------------------------------------------*/
int ebioskey(int a)
{
  int k;

  k=bioskey(a);
  return k;
}


/*-------------------------------------------------------*/
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

```
void init_system(void)
{
  int i;

  for(i=0;i<TOOLMAX;i++)
  {
    toolsize[i]=2.0;
    tooldeep[i]=0.0;
    objectdeep[i]=0.0;
    toolno[i]=1;
  }
}

/*-------------------------------------------------*/
void init_menu(void)
{
  dn_sel=(int *)malloc(sizeof(int));
  for (i=1;i<MAXMAINMENU+1;i++)
    pop_menu[i]=(PULLDOWN *)malloc(sizeof(PULLDOWN));
  for (i=1;i<MAXMAINMENU+1;i++)
    row[i]=1;

  pop_menu[1]->l=2;
  pop_menu[1]->t=MENU_H+2;
  pop_menu[1]->r=pop_menu[1]->l+textwidth("W")*30+20;
  pop_menu[1]->max=4;
  pop_menu[1]->b=pop_menu[1]->t+ (pop_menu[1]->max)*MENU_H+2;
  pop_menu[1]->text=" บันทึกข้อมูล Project ลงแผ่นบันทึก \n อ่านข้อมูล Project จากแผ่นบันทึก \n อ่านข้อมูลภาพด้านบน dxf file \n ออกจาก
โปรแกรม \n";
  pop_menu[1]->help=" file2222222 \n 33333program \n 122333333323 \n 456 \n";
  pop_menu[1]->save=NULL;

  pop_menu[2]->l=50;
  pop_menu[2]->t=MENU_H+2;
  pop_menu[2]->r=pop_menu[2]->l+textwidth("W")*15+20;
  pop_menu[2]->max=2;
  pop_menu[2]->b=pop_menu[2]->t+ (pop_menu[2]->max)*MENU_H+2;
  pop_menu[2]->text=" ชิ้นงานใหม่ \n เครื่องมือ \n";
  pop_menu[2]->help=" 123456fil \n 456789progra \n";
  pop_menu[2]->save=NULL;

  pop_menu[3]->l=170;
  pop_menu[3]->t=MENU_H+2;
  pop_menu[3]->r=pop_menu[3]->l+textwidth("W")*26+20;
  pop_menu[3]->max=5;
  pop_menu[3]->b=pop_menu[3]->t+ (pop_menu[3]->max)*MENU_H+2;
  pop_menu[3]->text=" กำหนดชั้นความลึกชิ้นงาน \n สร้างโปรแกรมควบคุมเครื่อง(Auto) \n แก้ไขความเร็ว(Feed Rate) \n บันทึกโปรแกรมควบคุม
เครื่อง \n อ่านโปรแกรมควบคุมเครื่อง \n";
  pop_menu[3]->help=" 987 \n 5645 \n 21212123 \n 0121545456 \n";
  pop_menu[3]->save=NULL;

  pop_menu[4]->l=320;
  pop_menu[4]->t=MENU_H+2;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  pop_menu[4]->r=pop_menu[4]->l+textwidth("W")*20+20;
  pop_menu[4]->max=3;
  pop_menu[4]->b=pop_menu[4]->t+ (pop_menu[4]->max)*MENU_H+2;
  pop_menu[4]->text=" เริ่มต้นจำลองการทำงาน \n รูปชิ้นงานหลังการกัด \n ภาพทับชิ้นงาน \n";
  pop_menu[4]->help=" 88877333337 \n 5554443334 \n 1111133331 \n";
  pop_menu[4]->save=NULL;

  pop_menu[5]->l=440;
  pop_menu[5]->t=MENU_H+2;
  pop_menu[5]->r=pop_menu[5]->l+textwidth("W")*18+20;
  pop_menu[5]->max=4;
  pop_menu[5]->b=pop_menu[5]->t+ (pop_menu[5]->max)*MENU_H+2;
  pop_menu[5]->text=" Feed Rate (ค่าป้อน) \n Spindle Rate \n Jog Speed \n หน่วย นิ้ว/มม. \n";
  pop_menu[5]->help=" 8887737 \n 5554234 \n 1111133331 \n 0333333 \n";
  pop_menu[5]->save=NULL;

  pop_menu[6]->l=500;
  pop_menu[6]->t=MENU_H+2;
  pop_menu[6]->r=pop_menu[6]->l+textwidth("W")*18+15;
  pop_menu[6]->max=3;
  pop_menu[6]->b=pop_menu[6]->t+ (pop_menu[6]->max)*MENU_H+2;
  pop_menu[6]->text=" เคลื่อนที่ X,Y,Z \n Home X,Y,Z \n Set Zero(Rel.Pos.) \n";
  pop_menu[6]->help=" 4444444 \n 3333333 \n 2222222222 \n";
  pop_menu[6]->save=NULL;

  pop_menu[7]->l=580;
  pop_menu[7]->r=pop_menu[7]->l+textwidth("W")*18+20;
  pop_menu[7]->t=MENU_H+2;
  pop_menu[7]->max=4;
  pop_menu[7]->b=pop_menu[7]->t+ (pop_menu[7]->max)*MENU_H+2;
  pop_menu[7]->text=" Cycle Start \n Cycle Stop \n Spindle On/Off \n Single Block \n";
  pop_menu[7]->help=" aaaaaaa \n sssssss \n qqqqqqqqqq \n wwwwwww \n";
  pop_menu[7]->save=NULL;
}

/*-------------------------------------------------------------*/
int readkbd(void)
{
  int ch;

  ch=getch();

  return(ch);
}

void dispblank_xy_text(int x,int y,int maxlen,int col)
{
  int i;
  char buf[250];

  for(i=0;i<maxlen;i++)
    buf[i]=' ';
  buf[i]=0;

  ¥
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

```
  drawtextbackground(x,y,buf,col,col,col);/*- dir -*/
}

void dispblank_text(int x,int y,int maxlen,int col)
{
  dispblank_xy_text(x*8,y*22,maxlen,col);
}


void dispblank(int x,int y,int maxlen,int attr)
{

/*  setfillstyle(SOLID_FILL,attr);
  bar(x,y,x+maxlen*8,y+20);
*/
  int i;
  char buf[250];

  for(i=0;i<maxlen;i++)
    buf[i]=' ';
  buf[i]=0;

  drawtextbackground(x*8,y*22,buf,attr,attr,attr);/*- dir -*/
}

int getstring(char textst[],int x,int y,int maxlen,int mode,int attr)
{
  int inkey;
  char key;
  int dot=OFF;

  dispblank(x,y,maxlen,GREEN);
  drawtextbackground(x*8,y*22,textst,LIGHTRED,BLUE,DARKGRAY);/*- dir -*/

  setcolor(attr);
  do{
    switch(mode)
    {
            case THAIENG   : inkey = readkbd();      break;
            case NUMBER    :
            case ONEORTWO  :
            case ENGLISH   : inkey = bioskey(0);     break;
            case ENGUPCASE : inkey = bioskey(0);
                                    inkey = readkbd();
                             key = inkey & 0xff;
                          if ((key >= 'a') && (key <= 'z'))
                            inkey = key - ('a' - 'A');
                          break;
    }

    switch(inkey)
    {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    case ESC    : textst[0]=0;
                  return(NO);


    case BSKEY  : if(strlen(textst)>0)
                  {
                    if(textst[strlen(textst)-1]=='.')
                        dot=OFF;
                    textst[strlen(textst)-1]='\0';
                  }
                  dispblank(x,y,maxlen,GREEN);
                  drawtextbackground(x*8,y*22,textst,LIGHTRED,BLUE,DARKGRAY);/*- dir -*/
                  break;

    case RETKEY : return(YES);


    default:
          inkey = (inkey & 0xff);

          if (mode == NUMBER) {
             if( ((inkey < '0') || (inkey > '9')) &&
                 (inkey!='.') && (inkey!='-')  ) break;
          } /*- end if mode NUMBER -*/
          if( (inkey=='.')&&(dot==OFF))
            dot=ON;
          else
            if(inkey=='.')
              inkey=0;
          if( (inkey >= 32)  )
          {
            if (strlen(textst) < maxlen)
            {
              textst[strlen(textst)+1] = '\0';
              textst[strlen(textst)] = inkey;
              dispblank(x,y,maxlen,GREEN);
              drawtextbackground(x*8,y*22,textst,LIGHTRED,BLUE,DARKGRAY);
            }
          }
   }
  }while(1);
}

/*---------------------- o b j e c t ----------------------*/
/*---------------------- 3 D  G r a p h ----------------------*/
/*---------------------- d x f  f i l e ----------------------*/

void init_dxf_data(void)
{
  int i,j;

  currentlayer.num=0;   /*-- number of picture in TOP --*/
  for(i=0;i<POLYMAX;i++)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
 {
   currentlayer.n[i]=0;
   currentlayer.type[i]=0;
   currentlayer.xmax[i]=0.0;
   currentlayer.ymax[i]=0.0;
   currentlayer.xmin[i]=0.0;
   currentlayer.ymin[i]=0.0;
   for(j=0;j<lMAX;j++)
   {
     currentlayer.poly[i].x[j]=0.0;
     currentlayer.poly[i].y[j]=0.0;
     currentlayer.poly[i].r[j]=0.0;
     currentlayer.poly[i].s[j]=0;
   }
 }
}

void pointto(double x,double y)
{
  moveto(x,y);
  linerel(2,2);
  linerel(-4,-4);
  linerel(2,2);
  linerel(-2,2);
  linerel(4,-4);
}

int open_dxf_file(char *fname)
{
  if( (datadxf=fopen(fname,"rb"))==NULL )
  {
    return(NO);
  }
  return(YES);
}

void trim(char *text)
{
  int i=0,j=0;
  char dest[255];

  while(text[i]<=32)
    i++;

  do{
    if(text[i]>=32)
    {
      dest[j]=text[i];
      j++;
    }
    i++;
  }while(text[i]!=0);
  dest[j]=0;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการ
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใ

```
  i=0;
  while(dest[i]!=0)
  {
   text[i]=dest[i];
    i++;
  }
  text[i]=0;
}

int search_entities(void)
{
  char text[255]; //,*tx=text;
  char txt[255];
  int found=-1;
  int a;
  int i,j;

  rewind(datadxf);
  do{
    fgets(text,255,datadxf);

   text[strlen(text)-1] = '\0';

    i=j=0;
    while(text[i]<=32)
      i++;
    while(text[i]!=0)
    {
      if(text[i]>=32)
      {
                txt[j]=text[i];
                j++;
      }
      i++;
    }
    txt[j]=0;
    if( strncmp(txt,"ENTITIES",8)==0 )
      found=1;
    a=feof(datadxf);
  } while( (!(a)) && (found!=1) );
  if(found==1)
    return(YES);
  else
    return(NO);
}

void trim00(char *text[])
{
  int i=0,j=0;
  char dest[255];

  while(text[0][i]<=32)
    i++;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  do{
    if(text[0][i]>=32)
    {
      dest[j]=text[0][i];
      j++;
    }
    i++;
  }while(text[0][i]!=0);
  dest[j]=0;
  strcpy(text[0],dest);
}

void readxyz_dxf(double *x,double *y,double *z,FILE *data)
{
  char text[255];

  do{
    fgets(text,255,data);
    trim(text);
  }while( strncmp(text,"10",2)!=0 );
  fgets(text,255,datadxf);  /*- read x -*/
  trim(text);
  *x=atof(text);

  do{
    fgets(text,255,data);
    trim(text);
  }while( strncmp(text,"20",2)!=0 );
  fgets(text,255,datadxf);  /*- read y -*/
    trim(text);
  *y=atof(text);

  do{
    fgets(text,255,data);
    trim(text);
  }while( strncmp(text,"30",2)!=0 );
  fgets(text,255,datadxf);  /*- read z -*/
  trim(text);
  *z=atof(text);
}

 void read_dxf_data(void)
 {
  int quit=OFF;
  char text[255],text1[255];
  double dxc,dyc,dzc,dr;
  double dxr,dyr,dzr;
  double dxs,dys,dzs;
  double dxbegin,dybegin;
  double cxs,cys; //,czs;

  double a,d,b,r;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
double xe,ye;
double xs,ys;
double xc,yc;
double k,m;
int sg=GRAPH;

GRAPH=OFF;


do{
 fgets(text,255,datadxf);
 trim(text);

 if( strncmp(text,"CIRCLE",6)==0 )
 {
  readxyz_dxf(&dxc,&dyc,&dzc,datadxf);

  do{
        fgets(text,255,datadxf);
        trim(text);
  }while( strncmp(text,"40",2)!=0 );

  fgets(text,255,datadxf);  /*- read x -*/
  trim(text);
  dr=atof(text);
  if(GRAPH==ON)
  {
        circle((int)(dxc),(int)(dyc),(int)(dr));
  }
  add_circle(dxc,dyc,dr);
 }

 if( strncmp(text,"POLYLINE",8)==0)
 {
  /*----- read x,y,z for reference point -----*/
  readxyz_dxf(&dxr,&dyr,&dzr,datadxf);
  fgets(text,255,datadxf);
  fgets(text,255,datadxf);

  init_poly(dxr,dyr);

  readxyz_dxf(&dxs,&dys,&dzs,datadxf);
  fgets(text,255,datadxf);
  fgets(text,255,datadxf);

  add_poly(dxs,dys,0.0);

  if(GRAPH==ON)
  {
        pointto(dxr+dxs,dyr+dys);
        moveto(dxr+dxs,dyr+dys);
  }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
dxbegin=dxr+dxs; dybegin=dyr+dys;

xs=dxbegin; ys=dybegin;

do{
        do{

          readxyz_dxf(&dxc,&dyc,&dzc,datadxf);

          if(GRAPH==ON)
          {
            lineto(dxr+dxc,dyr+dyc);
          }
          cxs=dxr+dxc; cys=dyr+dyc;
          b=0.0;
          xe=cxs;  ye=cys;

          fgets(text1,255,datadxf);
          trim(text1);
          if(strncmp(text1,"42",2)==0 )
          {
            fgets(text1,255,datadxf);
            text1[strlen(text1)-1] = '\0';
            b=atof(text1);
            fgets(text1,255,datadxf);


            if(GRAPH==ON)
            {
              setcolor(LIGHTGREEN);
              line(xs,ys,xe,ye);
            }

            a=4*atan(b);
            d=pow( ( (xs-xe)*(xs-xe) + (ye-ys)*(ye-ys) ),0.5 );
            r = d/(2*sin(a/2));
            if(ys!=ye)
            {
              k=(xs*xs + ys*ys - xe*xe -ye*ye)/( 2*(ys-ye));
              m=(xs-xe)/(ys-ye);
              xc=-((ys-k)*m-xs)/(1-m*m) -  pow (( pow(((ys-k)*m-xs),2) - (1-m*m)*(xs*xs + pow((ys-k),2)-dr*r) ),(0.5) )/(1-m*m);
              yc=(xs*xs + ys*ys -xe*xe - ye*ye -2*(xs-xe)*xc)/(2*(ys-ye));
            }else
            {
              xc=(xs+xe)/2.0;
              m=fabs(xs-xc);
              k=pow(m*m + r*r ,0.5);
              yc=xs-k;
            }
            if(GRAPH==ON)
            {
              putpixel(xc,yc,LIGHTGREEN);
              setcolor(YELLOW);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            line(xc,yc,xs,ys);
            line(xc,yc,xe,ye);
          }

          /*-          d2=360-atan2(ys-yc,xs-xc)*180.0/M_PI; -*/
          /*-          d1=360-atan2(ye-yc,xe-xc)*180.0/M_PI; -*/

          r=pow( (ys-yc)*(ys-yc) + (xs-xc)*(xs-xc) ,0.5);

          /*-          arc(xc,yc, d1 , d2 ,r); -*/

          setcolor(RED);
        } else
        {
          setcolor(WHITE);
        }

        xs=xe;  ys=ye;

        if(GRAPH==ON)
        {
          lineto(dxr+dxc,dyr+dyc);
        }

        add_poly(dxc,dyc,b);

        fgets(text,255,datadxf);
        text[strlen(text)-1] = '\0';
        trim(text);
      } while( (strncmp(text,"SEQEND",6)!=0) && (strncmp(text,"VERTEX",6)!=0) );
      if( strncmp(text,"SEQEND",6)==0)
      {
        if(GRAPH==ON)
        {
          lineto(dxbegin,dybegin);
        }
        add_poly(dxbegin,dybegin,b);
      }
    } while( strncmp(text,"SEQEND",6)!=0 );

  }
  if( strncmp(text,"ENDSEC",6)==0 )
    quit=ON;
  if( strncmp(text,"EOF",3)==0)
    quit=ON;
 }while(quit==OFF);
 GRAPH=sg;
}

void add_circle(double x,double y,double r)
{
 int n=currentlayer.num;
 // struct pic *ln;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  currentlayer.n[n]=1;
  currentlayer.type[n]=CIRCLE;
  currentlayer.poly[n].x[0]=x;
  currentlayer.poly[n].y[0]=y;
  currentlayer.poly[n].r[0]=r;
  currentlayer.poly[n].s[0]=-1;
  currentlayer.num++;
}

void init_poly(double x,double y)
{
  int n=currentlayer.num;
  // struct pic *ln;

  currentlayer.n[n]=1;
  currentlayer.type[n]=POLYLINE;
  currentlayer.poly[n].x[0]=x;
  currentlayer.poly[n].y[0]=y;
  currentlayer.poly[n].r[0]=0.0;
  currentlayer.poly[n].s[0]=-1;
  currentlayer.num++;
}

void add_poly(double x,double y,double r)
{
  int n=currentlayer.num-1;
  int i=currentlayer.n[n];

  if(i>IMAX) {
    return;
  }
  currentlayer.n[n]++;
  currentlayer.poly[n].x[i]=x;
  currentlayer.poly[n].y[i]=y;
  currentlayer.poly[n].r[i]=r;
  currentlayer.poly[n].s[i]=-1;
}

void calc_dxf_data(void)
{
  int i,j;
  double xr,yr,xs,ys,x,y,r;

  for(i=0;i<POLYMAX;i++)
  {
    if(currentlayer.n[i]>0)
    {
      switch(currentlayer.type[i])
      {
            case CIRCLE   :
                  x=currentlayer.poly[i].x[0];
                  y=currentlayer.poly[i].y[0];
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                    r=currentlayer.poly[i].r[0];

                    currentlayer.xmax[i]=x+r;
                    currentlayer.ymax[i]=y+r;
                    currentlayer.xmin[i]=x-r;
                    currentlayer.ymin[i]=y-r;
                break;
            case POLYLINE :
                    currentlayer.xmax[i]=currentlayer.ymax[i]=0.0;
                    currentlayer.xmin[i]=currentlayer.ymin[i]=10000.0;

                    xr=currentlayer.poly[i].x[0];
                    yr=currentlayer.poly[i].y[0];

                    xs=currentlayer.poly[i].x[1];
                    ys=currentlayer.poly[i].y[1];

                    if(xr+xs>currentlayer.xmax[i])currentlayer.xmax[i]=xr+xs;
                    if(yr+ys>currentlayer.ymax[i])currentlayer.ymax[i]=yr+ys;
                    if(xr+xs<currentlayer.xmin[i])currentlayer.xmin[i]=xr+xs;
                    if(yr+ys<currentlayer.ymin[i])currentlayer.ymin[i]=yr+ys;


                    for(j=2;j<currentlayer.n[i];j++)
                    {
                      x=currentlayer.poly[i].x[j];
                      y=currentlayer.poly[i].y[j];

                      if(xr+x>currentlayer.xmax[i])currentlayer.xmax[i]=xr+x;
                      if(yr+y>currentlayer.ymax[i])currentlayer.ymax[i]=yr+y;
                      if(xr+x<currentlayer.xmin[i])currentlayer.xmin[i]=xr+x;
                      if(yr+y<currentlayer.ymin[i])currentlayer.ymin[i]=yr+y;
                    }
                break;
      }
      //    currentlayer.xmax[i]+=toolr;    currentlayer.ymax[i]-=toolr;
      //    currentlayer.xmin[i]-=toolr;    currentlayer.ymin[i]-=toolr;
    }
  }
}

void draw_dxf_3d(void)
{
  int i,j;

  double  x1, y1, x2, y2;
  double sx1,sy1,sx2,sy2;
  double xr,yr,xs,ys,x,y,r,z;

  setcolor(COL_DXF3D_OBJ);
  for(i=0;i<POLYMAX;i++)
  {
    if(currentlayer.n[i]>=0)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{
 switch(currentlayer.type[i])
 {
        case CIRCLE   :
                    x=currentlayer.poly[i].x[0];
                    y=currentlayer.poly[i].y[0];
                    r=currentlayer.poly[i].r[0];

                    x=x-dxfx1; y=y-dxfy1;
                    if(i>0)
                    {
                     if(objectdeep[i-1]>0.0)
                     {
                       z=-objectdeep[i-1];
                       circle_z3d(x,y,z  ,r,COL_DXF3D_OBJ);
                       circle_z3d(x,y,0.0,r,COL_DXF3D_OBJ);

                       line3df(x+r,y  ,0.0,x+r,y  ,z,COL_DXF3D_LINE);
                       line3df(x-r,y  ,0.0,x-r,y  ,z,COL_DXF3D_LINE);
                       line3df(x  ,y+r,0.0,x  ,y+r,z,COL_DXF3D_LINE);
                       line3df(x  ,y-r,0.0,x  ,y-r,z,COL_DXF3D_LINE);
                     }
                     else
                       circle_z3d(x,y,0.0,r,COL_DXF3D_OBJ);
                    }
            break;
        case POLYLINE :

                    xr=currentlayer.poly[i].x[0];
                    yr=currentlayer.poly[i].y[0];

                    xs=currentlayer.poly[i].x[1];
                    ys=currentlayer.poly[i].y[1];

                    x1=xr+xs;  y1=yr+ys;

                    for(j=2;j<currentlayer.n[i];j++)
                    {
                       x=currentlayer.poly[i].x[j];
                       y=currentlayer.poly[i].y[j];
                       x2=xr+x;  y2 = yr+y;

                       sx1 = x1 - dxfx1;  sy1=y1-dxfy1;
                       sx2 = x2 - dxfx1;  sy2=y2-dxfy1;


                       if(i>0)
                       {
                        if(objectdeep[i-1]>0.0)
                        {
                          line3df(sx1,sy1,0.0,
                                  sx2,sy2,0.0,COL_DXF3D_OBJ);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                    z=-objectdeep[i-1];
                    line3df(sx1,sy1,z  , sx2,sy2,z,COL_DXF3D_OBJ);

                    line3df(sx1,sy1,0.0,sx1,sy1,z,COL_DXF3D_LINE);
                    line3df(sx2,sy2,0.0,sx2,sy2,z,COL_DXF3D_LINE);
                   }else
                   {
                    line3df(sx1,sy1,0.0,sx2,sy2,0.0,COL_DXF3D_OBJ);
                   }
                  }else
                  {
                    line3df(sx1,sy1,0.0 ,sx2,sy2,0.0 ,WHITE);

                    z=-objectheight;

                    line3df(sx1,sy1,z   ,sx2,sy2,z   ,WHITE);
                    line3df(sx1,sy1,0.0 ,sx1,sy1,z   ,COL_DXF3D_LINE);
                    line3df(sx2,sy2,0.0 ,sx2,sy2,z   ,COL_DXF3D_LINE);
                  }

                  x1=x2;  y1=y2;
                 }
            break;
     }
    }
   }
}

void draw_dxf_data(void)
{
  int i,j;
  int fillcol=1;
  int suby=getmaxy();

  double xr,yr,xs,ys,x,y,r;

  setcolor(WHITE);
  for(i=0;i<POLYMAX;i++)
  {
   if(currentlayer.n[i]>0)
   {
    switch(currentlayer.type[i])
    {
          case CIRCLE   :
                  x=currentlayer.poly[i].x[0];
                  y=currentlayer.poly[i].y[0];
                  r=currentlayer.poly[i].r[0];
                  setcolor(WHITE);
                  circle(x,suby-y,r);
                  setfillstyle(SOLID_FILL,fillcol);
                  fillcol++;
                  floodfill(x,suby-y,WHITE);
            break;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            case POLYLINE :

                        xr=currentlayer.poly[i].x[0];
                        yr=currentlayer.poly[i].y[0];

                        xs=currentlayer.poly[i].x[1];
                        ys=currentlayer.poly[i].y[1];

                        /*-pointto(xr+xs,yr+ys);-*/
                        setcolor(WHITE);
                        moveto(xr+xs,suby-(yr+ys));

                        for(j=2;j<currentlayer.n[i];j++)
                        {
                          x=currentlayer.poly[i].x[j];
                          y=currentlayer.poly[i].y[j];
                          lineto(xr+x,suby-(yr+y));
                        }
                        setfillstyle(SOLID_FILL,fillcol);
                        fillcol++;
                        x=(currentlayer.xmax[i]+currentlayer.xmin[i])/2.0;
                        y=(currentlayer.ymax[i]+currentlayer.ymin[i])/2.0;
                        floodfill(x,suby-y,WHITE);
                        /*- lineto(xr+xs,yr+ys); -*/
                    break;
      }
     }
    }
}


void calc_dxf_max_data(void)
{
  int i;
  double max,xmax,ymax,xmin,ymin;
  double xtemp,ytemp,ntemp,ttemp,rtemp;
  //  struct pic pt;


  xmax=currentlayer.xmax[0];
  xmin=currentlayer.xmin[0];

  ymax=currentlayer.ymax[0];
  ymin=currentlayer.ymin[0];

  max=0;

  for(i=1;i<POLYMAX;i++)
  {
    if( (xmax<currentlayer.xmax[i])&&(xmin>currentlayer.xmin[i])
     && (ymax<currentlayer.ymax[i])&&(ymin>currentlayer.ymin[i]))
    {
      xmax=currentlayer.xmax[i];   xmin=currentlayer.xmin[i];
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    ymax=currentlayer.ymax[i];    ymin=currentlayer.ymin[i];
    max=i;
  }
}
xtemp=currentlayer.xmax[0];
currentlayer.xmax[0]=xmax;
currentlayer.xmax[max]=xtemp;

ytemp=currentlayer.ymax[0];
currentlayer.ymax[0]=ymax;
currentlayer.ymax[max]=ytemp;

xtemp=currentlayer.xmin[0];
currentlayer.xmin[0]=xmin;
currentlayer.xmin[max]=xtemp;

ytemp=currentlayer.ymin[0];
currentlayer.ymin[0]=ymin;
currentlayer.ymin[max]=ytemp;

ntemp=currentlayer.n[0];
currentlayer.n[0]=currentlayer.n[max];
currentlayer.n[max]=ntemp;

ttemp=currentlayer.type[0];
currentlayer.type[0]=currentlayer.type[max];
currentlayer.type[max]=ttemp;

for(i=0;i<IMAX;i++)
{
  xtemp=currentlayer.poly[0].x[i];
  currentlayer.poly[0].x[i]=currentlayer.poly[max].x[i];
  currentlayer.poly[max].x[i]=xtemp;

  ytemp=currentlayer.poly[0].y[i];
  currentlayer.poly[0].y[i]=currentlayer.poly[max].y[i];
  currentlayer.poly[max].y[i]=ytemp;

  rtemp=currentlayer.poly[0].r[i];
  currentlayer.poly[0].r[i]=currentlayer.poly[max].r[i];
  currentlayer.poly[max].r[i]=rtemp;
}

dxfx1=currentlayer.xmin[0];
dxfx2=currentlayer.xmax[0];

dxfy1=currentlayer.ymin[0];
dxfy2=currentlayer.ymax[0];

dxfy=dxfy2-dxfy1;
dxfx=dxfx2-dxfx1;

// setcolor(LIGHTBLUE);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  // rectangle(xmin,ymin,xmax,ymax);
}

void calcX2line(double x1,double y1,double x2,double y2,
                double x3,double y3,double x4,double y4,
                double *x,double *y)
{
  double m1,m2,c1,c2;

  if((x1!=x2)&&(x3!=x4))
  {
    m1=(y2-y1)/(x2-x1);
    c1=y1-m1*x1;
    m2=(y4-y3)/(x4-x3);
    c2=y3-m2*x3;
    *x=(c2-c1)/(m1-m2);
    *y=*x*m1 +c1;
  }else
  {
    if( (x1==x2)&&(x4!=x3) )
    {
      *x=x1;
      m1 = (y4-y3)/(x4-x3);
      c2 = y4 - m1*x4;
      *y = *x*m1 + c2;
    }
    else
    {
      *x=x3;
      m1 = (y2-y1)/(x2-x1);
      c2 = y2 - m1*x2;
      *y = *x*m1 + c2;
    }
  }
}

void draw_dxf_fill(int mode)
{
  struct pic_offset offset[50];
  struct point cross[50];
  char text[256];

  FILE *ofs;

  int suby=maxy;
  int COL=LIGHTGRAY;
  int n;
  int ct;
  int ii;
  int fillcol=1;
  int status,count;
  int n_offset;
  int ncross;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
double yl;
double i,j;
double x,y,r,xr,yr,xs,ys;
double x1,y1,x2,y2;
double m,c;
double x3,y3;
double xx1s,yy1s,xxs,yys,sxx,syy;
double xx,yy,yyt,xxt;
double xx1,yy1,xx2,yy2;
double xx3,yy3,xx4,yy4;
double sxx1,syy1,sxx2,syy2;
double sxx3,syy3,sxx4,syy4;
double xmax,ymax,xmin,ymin;
double deg,tootx,tooly;
double zcut;
double savetools=toolr;


switch(mode)
{
  case SHOWDXF:   toolr=1.0;              break;
  case SHOWTEST:  toolr=5.0;              break;
  case SHOWCUT:   toolr=toolsize[toolno[0]]; break;
}

if( (ofs=fopen(OFFSETNAME,"wt"))==NULL )
{
  printf("can't open offset file for save");
  exit(0);
}
if( (datacut=fopen(CUTNAME,"wt"))==NULL )
{
  printf("can't open cut file for save");
  exit(0);
}

setcolor(COL);
for(n=1;n<POLYMAX;n++)
{
  if(currentlayer.n[n]>0)
  {
    switch(currentlayer.type[n])
    {
          case CIRCLE   :

                    if(mode==SHOWCUT)
                    {
                      if(objectdeep[n-1]==0.0)
                        break;
                      else
                        toolr = toolsize[toolno[n-1]];
                    }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
zcut=objectdeep[n-1];
x=currentlayer.poly[n].x[0];
y=currentlayer.poly[n].y[0];
r=currentlayer.poly[n].r[0];
c=r-toolr;
xmin=x-c;        xmax=x+c;
ymin=y-r+toolr;  ymax=y+r-toolr;

switch(mode){
  case SHOWDXF:
    setcolor(WHITE);
    Circledxf(x,y,r);
  break;
  case SHOWTEST:
    setcolor(WHITE);
    circle(x,suby-y,r);
  break;
  case SHOWCUT:
    fprintf(datacut,"C %016.12f %016.12f %016.12f %016.12f\n"
                                  ,x-dxfx1,y-dxfy1,zcut,r);
  break;
}
fprintf(ofs,"C %016.12f %016.12f %016.12f\n",x,y,r);

if( (objectdeep[n-1]>=objectheight)&&(mode==SHOWCUT) )
  break;

for(yl=ymin; yl<=ymax ;yl+=toolr)
{
  m=yl-y;
  m=floor(fabs(c*c-m*m)*1.0e12)/1.0e12;
  xx=sqrt(m);

  switch(mode)
  {
    case SHOWDXF:
            setcolor(fillcol);
            Linedxf(x+xx,yl,x-xx,yl);
    break;

    case SHOWTEST:
            setcolor(LIGHTGREEN);
            circle(x+xx,suby-yl,toolr);
            circle(x-xx,suby-yl,toolr);
            setcolor(WHITE);
            line(x+xx,suby-yl,x-xx,suby-yl);
    break;
    case SHOWCUT:
            fprintf(datacut,"L %016.12f %016.12f %016.12f %016.12f %016.12f %016.12f\n",
                    x+xx-dxfx1,yl-dxfy1,zcut,x-xx-dxfx1,yl-dxfy1,zcut);
    break;
  }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการ

```
}
/*
/*
/*
/*
/*------
for(j=cu                                    ...j,j+=toolr)
{
}          ...ayer.poly[n].x[0];
yr=currentlayer.poly[n].y[0];
xs=currentlayer.poly[n].x[1];
ys=currentlayer.poly[n].y[1];

x1=xr+xs;  y1=yr+ys;
for(ii=2;ii<currentlayer.n[n];ii+=1)
{
 x=currentlayer.poly[n].x[ii];
 y=currentlayer.poly[n].y[ii];

 x2=xr+x; y2=yr+y;

 xx = (x1+x2)/2;
 yy = (y1+y2)/2;

 switch(mode)
 {
  case SHOWTEST:
          setcolor(YELLOW);
          line(x1,suby-y1,x2,suby-y2);
          setcolor(MAGENTA);
          line(x1,suby-y1,x2,suby-y2);
          setcolor(LIGHTRED);
          circle(xx,suby-yy,3);
          line(xx,suby-yy,0,suby-yy);
  break;
 }

 if( y1 == y2 ) {
  count = 1;
  xx1=xr+xs;  yy1=yr+ys;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
for(ct=2;ct<currentlayer.n[n];ct+=1)
{
        x=currentlayer.poly[n].x[ct];
        y=currentlayer.poly[n].y[ct];
        xxs = xx2 = xr + x;
        yys = yy2 = yr + y;

        if(xx1>xx2){
          yyt=yy1;  yy1=yy2;  yy2=yyt;
          xxt=xx1;  xx1=xx2;  xx2=xxt;
        }
        if( (xx1<=xx) && (xx<=xx2) )
        {
          if(xx2!=xx1)
          {
            m=(yy2-yy1)/(xx2-xx1);
            c=yy1-m*xx1;
            y=m*xx+c;
            if( (y-yy)>1.0e-12 )
            {
                    switch(mode)
                    {
                    case SHOWTEST:
                            setcolor(GREEN);
                            circle(xx,suby-y,2);
                    break;
                    }
              count++;
            }
          } else {
            if(xx1==xx)
            {
              if(yy1>yy)
              {
                    switch(mode)
                    {
                      case SHOWTEST:
                              setcolor(GREEN);
                              circle(xx,suby-y,2);
                      break;
                    }
                    count++;
              }
            }
          }
        }
        yy1=yys;xx1=xxs;
  } /*-- end for ct --*/
 } else {
  count = 0;
  xx1=xr+xs;  yy1=yr+ys;
  for(ct=2;ct<currentlayer.n[n];ct+=1)
  {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                x=currentlayer.poly[n].x[ct];
                y=currentlayer.poly[n].y[ct];

                xxs = xx2 = xr + x;
                yys = yy2 = yr + y;

                if(yy1>yy2){
                  yyt=yy1;  yy1=yy2;  yy2=yyt;
                  xxt=xx1;  xx1=xx2;  xx2=xxt;
                }
                if( (yy1<=yy) && (yy<=yy2) )
                {
                  if(yy2!=yy1)
                  {
                    x=(yy-yy1)*(xx1-xx2)/(yy1-yy2) + xx1;
                    if( (x-xx)<1.0e-12 )
                    {
                            switch(mode)
                            {
                              case SHOWTEST:
                                      setcolor(GREEN);
                                      circle(xx,suby-y,2);
                            break;
                            }
                      count++;
                    }
                  } else {
                    if(yy1==yy)
                    {
                      if(xx1<xx)
                      {
                            switch(mode)
                            {
                              case SHOWTEST:
                                      setcolor(GREEN);
                                      circle(xx,suby-y,2);
                            }
                            count++;
                      }
                    }
                  }
                }
                yy1=yys;xx1=xxs;
              } /*-- end for ct --*/
            } /*- end else if (y1==y2) -*/
            if(count%2 == 0)
              currentlayer.poly[n].s[ii]=0;
            else
              currentlayer.poly[n].s[ii]=1;
            x1=x2;
            y1=y2;
          } /*-  end for ii  -*/
        } /*-  end for j  -*/
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
/*-----------------------------------------*/
/*                                         */
/*      D R A W   P O L Y L I N E          */
/*                                         */
/*-----------------------------------------*/
xr=currentlayer.poly[n].x[0];
yr=currentlayer.poly[n].y[0];
xs=currentlayer.poly[n].x[1];
ys=currentlayer.poly[n].y[1];

x1=xr+xs;  y1=yr+ys;
/*-cleardevice();-*/
for(ii=2;ii<currentlayer.n[n];ii+=1)
{
        x=currentlayer.poly[n].x[ii];
        y=currentlayer.poly[n].y[ii];

        x2=xr+x; y2=yr+y;
        switch(mode)
        {
          case SHOWDXF:
                setcolor(WHITE);
                Linedxf(x1,y1,x2,y2);
          break;
          case SHOWTEST:
                setcolor(WHITE);
                line(x1,suby-y1,x2,suby-y2);
          break;
        }
        x1=x2;  y1=y2;
}
/*-----------------------------------------*/
/*                                         */
/*      Section LINE OFFSET                */
/*                                         */
/*-----------------------------------------*/
/*-cleardevice();-*/
xr=currentlayer.poly[n].x[0];
yr=currentlayer.poly[n].y[0];
xs=currentlayer.poly[n].x[1];
ys=currentlayer.poly[n].y[1];

x1=xr+xs;           y1=yr+ys;
x=currentlayer.poly[n].x[2];
y=currentlayer.poly[n].y[2];
x2=xr+x;                    y2=yr+y;

ii=0;
for(j=3;j<=currentlayer.n[n];j+=1)
{
        if(j!=currentlayer.n[n])
        {
          x=currentlayer.poly[n].x[j];
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  y=currentlayer.poly[n].y[j];
} else
{
  x=currentlayer.poly[n].x[2];
  y=currentlayer.poly[n].y[2];
}
x3=xr+x; y3=yr+y;

if(x2!=x1)
{
  deg=atan2(y2-y1,x2-x1);
  toolx=fabs(floor((toolr*sin(deg))*1.0e12)/1.0e12);
  tooly=fabs(floor((toolr*cos(deg))*1.0e12)/1.0e12);
} else {
  toolx=toolr;
  tooly=0;
}

if( (currentlayer.poly[n].s[j-1]) == 1 )
{
  xx1  = x1 + toolx;
  xx2  = x2 + toolx;

  sxx1 = x1 - toolx;
  sxx2 = x2 - toobx;

  if(x1!=x2)
    m=(y2-y1)/(x2-x1);

  if(m<1.0e-12)
    tooly=+tooly;
  else
    tooly=-tooly;

  yy1  = y1 + tooly;
  yy2  = y2 + tooly;

  syy1 = y1 - tooly;
  syy2 = y2 - tooly;
} else   /*-------- status=0 ----------*/
{
  xx1  = x1 - toolx;
  xx2  = x2 - toolx;
  sxx1 = x1 + toolx;
  sxx2 = x2 + toolx;

  if(x1!=x2)
    m=(y2-y1)/(x2-x1);

  if(m<1.0e-12)
    tooly=-tooly;
  else
    tooly=+tooly;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  yy1  = y1 + tooly;
  yy2  = y2 + tooly;
  syy1 = y1 - tooly;
  syy2 = y2 - tooly;
}

if(x3!=x2)
{
  deg=atan2(y3-y2,x3-x2);
  toolx=fabs(floor((toolr*sin(deg))*1.0e12)/1.0e12);
  tooly=fabs(floor((toolr*cos(deg))*1.0e12)/1.0e12);
} else {
  toolx=toolr;
  tooly=0;
}

if(j!=currentlayer.n[n])
  status=currentlayer.poly[n].s[j];
else
  status=currentlayer.poly[n].s[2];

if( status == 1 )
{
  xx3  = x2 + toolx;
  xx4  = x3 + toolx;
  sxx3 = x2 - toolx;
  sxx4 = x3 - toolx;

  if(x2!=x3)
    m=(y3-y2)/(x3-x2);

  if(m<1.0e-12)
    tooly=+tooly;
  else
    tooly=-tooly;

  yy3  = y2 + tooly;
  yy4  = y3 + tooly;
  syy3 = y2 - tooly;
  syy4 = y3 - tooly;

 } else
 {     /*-------- status = 0 ----------*/
  xx3  = x2 - toolx;
  xx4  = x3 - toolx;
  sxx3 = x2 + toolx;
  sxx4 = x3 + toolx;

  if(x2!=x3)
    m=(y3-y2)/(x3-x2);

  if(m<1.0e-12)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            tooly=-tooly;
          else
            tooly=+tooly;

          yy3  = y2 + tooly;
          yy4  = y3 + tooly;
          syy3 = y2 - tooly;
          syy4 = y3 - tooly;

        }
        calcX2line( xx1, yy1, xx2, yy2, xx3, yy3, xx4, yy4, &xx, &yy);
        calcX2line(sxx1,syy1,sxx2,syy2,sxx3,syy3,sxx4,syy4,&sxx,&syy);

        if(j==3)
        {
          fprintf(ofs,"L %016.12f %016.12f ",sxx,syy);
          xx1s=xxs=sxx;
          yy1s=yys=syy;
        }else{
          fprintf(ofs,"%016.12f %016.12f \n",sxx,syy);
          fprintf(ofs,"L %016.12f %016.12f ",sxx,syy);
        }
        switch(mode)
        {
          case SHOWTEST:  /*-  offset -*/
                    setcolor(RED);
                    line(xx1,suby-yy1,xx2,suby-yy2);
                    line(xx3,suby-yy3,xx4,suby-yy4);
                    circle((xx2+xx1)/2.0,suby-(yy2+yy1)/2,toolr);
                    circle((xx4+xx3)/2.0,suby-(yy4+yy3)/2,toolr);
                    circle(xx,suby-yy,toolr);

                    getch();

                    setcolor(GREEN);
                    line(sxx1,suby-syy1,sxx2,suby-syy2);
                    line(sxx3,suby-syy3,sxx4,suby-syy4);
                    circle((sxx2+sxx1)/2.0,suby-(syy2+syy1)/2,toolr);
                    circle((sxx4+sxx3)/2.0,suby-(syy4+syy3)/2,toolr);
                    circle(sxx,suby-syy,toolr);
            break;
        }
        offset[ii].x=xx;
        offset[ii].y=yy;
        offset[ii].s=currentlayer.poly[n].s[j-1];
        ii++;
        x1=x2;   y1=y2;
        x2=x3;   y2=y3;
    } /*- end for j -*/
    fprintf(ofs,"%016.12f %016.12f \n",xx1s,yy1s);

    offset[ii].x=offset[0].x;
    offset[ii].y=offset[0].y;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

```
offset[ii].s=offset[0].s;
n_offset=ii;

/*-------------------------------------------*/
/*        DRAW OFFSET          */
/*-------------------------------------------*/
switch(mode)
{
        case SHOWCUT:

            zcut = objectdeep[n-1];
            for(i=1;i<=n_offset;i+=1)
            {
                fprintf(datacut,"L %016.12f %016.12f %016.12f %016.12f %016.12f %016.12f \n",
                  offset[i].x  -dxfx1,offset[i].y  -dxfy1,zcut,
                  offset[i-1].x-dxfx1,offset[i-1].y-dxfy1,zcut );
            }
        break;
        case SHOWTEST:

            for(i=1;i<=n_offset;i+=1)
            {
                setcolor(LIGHTMAGENTA);
                line( offset[i].x  ,suby-offset[i].y,
                    offset[i-1].x,suby-offset[i-1].y);
                setcolor(MAGENTA);
                circle(offset[i].x  ,suby-offset[i].y,toolr);
                circle(offset[i-1].x,suby-offset[i-1].y,toolr);
            }
        break;
}

if( (mode==SHOWCUT)&&(objectdeep[n-1]>=objectheight) )
        break;

/*-------------------------------------*/
/*                                     */
/*    FILL POLYLINE                    */
/*                                     */
/*-------------------------------------*/
xmax=xmin=offset[0].x;
ymax=ymin=offset[0].y;
for(i=0;i<n_offset;i++)
{
        if(xmax<offset[i].x) xmax=offset[i].x;
        if(ymax<offset[i].y) ymax=offset[i].y;
        if(xmin>offset[i].x) xmin=offset[i].x;
        if(ymin>offset[i].y) ymin=offset[i].y;
}

if(mode==SHOWDXF)
        setcolor(fillcol);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
for(yl=ymin;yl<=ymax;yl+=toolr)
{
        x1=offset[0].x;
        y1=offset[0].y;
        ncross=0;
        for(i=0;i<50;i++)
        {
          cross[i].x=0;  cross[i].y=0;
        }
        for(i=1;i<=n_offset;i++)
        {
          x2=offset[i].x;
          y2=offset[i].y;

          /*--- find (xx,yy) cross line(x1,y1,x2,y2)  ---*/
          if( ((y1<=yl)&&(yl<=y2))||((y1>=yl)&&(yl>=y2)) )
          {

            if(y1==y2)
            {
              cross[ncross].x=x1;  cross[ncross].y=y1;
              ncross++;
              cross[ncross].x=x2;  cross[ncross].y=y2;
              ncross++;
            }else
            {
              if(x1==x2)
              {
                      xx=x1;
                      yy=yl;
                      cross[ncross].x=xx;  cross[ncross].y=yy;
                      ncross++;
              }else{
                      m=(y2-y1)/(x2-x1);
                      c=y1-m*x1;
                      yy=yl;
                      xx=(yy-c)/m;
                      cross[ncross].x=xx;  cross[ncross].y=yy;
                      ncross++;
              } /*- end if -*/
            } /*- end if -*/
          } /*- end if -*/
          x1=x2;y1=y2;
        } /*- end for i -*/
        /*------ sort cross --------*/
        ii=ncross;
        for(i=0;i<ncross;i++)
        {
          if( (cross[i].x!=0.0)&&(cross[i].y!=0.0) )
          {
            for(j=0;j<ncross;j++)
            {
              if((i!=j)&&(fabs(cross[i].x-cross[j].x)<1.0e-10)&&
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                    (fabs(cross[i].y-cross[j].y)<1.0e-10)   )
    {
        cross[j].x=0.0;cross[j].y=0.0;
        ii=ii-1;
    }
   }
  }
 }
ncross=ii;
if(ncross>1)
{
 for(i=ncross-1;i>0;i--)
 {
  for(j=0;j<i;j++)
  {
   if(cross[j].x<cross[j+1].x)
   {
        xx=cross[j].x;
        cross[j].x=cross[j+1].x;
        cross[j+1].x=xx;
        yy=cross[j].y;
        cross[j].y=cross[j+1].y;
        cross[j+1].y=yy;
   }
  }
 }
 /*-------------------------*/
 /*------ draw line --------*/
 /*-------------------------*/
 switch(mode)
 {
  case SHOWCUT:
        for(i=1;i<ncross;i+=2)
        {
         fprintf(datacut,"L %016.12f %016.12f %016.12f %016.12f %016.12f %016.12f \n",
                cross[i].x  -dxfx1 ,cross[i].y  -dxfy1,zcut,
                cross[i-1].x-dxfx1 ,cross[i-1].y-dxfy1,zcut);
        }
  break;
  case SHOWDXF:
        for(i=1;i<ncross;i+=2)
        {
         Linedxf( cross[i].x   ,cross[i].y,
                cross[i-1].x ,cross[i-1].y);
        }
  break;
  case SHOWTEST:

        for(i=1;i<ncross;i+=2)
        {
         setcolor(WHITE);
         line( cross[i].x   ,suby-cross[i].y,
                cross[i-1].x ,suby-cross[i-1].y);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                                    setcolor(LIGHTGREEN);
                                    circle(cross[i].x  ,suby-cross[i].y  ,toolr);
                                    circle(cross[i-1].x,suby-cross[i-1].y,toolr);
                                   }
                              break;
                             }
                            } /*-- end if (ncross>1) --*/
                  } /*- end for yl -*/
                   if(mode==SHOWTEST)
                         getch();
                  if(mode==SHOWDXF)
                         fillcol++;
                 break;
    } /*-- end case polyline --*/
  }
}
fclose(ofs);

xmin=xmax=currentlayer.poly[0].x[1];
ymin=ymax=currentlayer.poly[0].y[1];
for(ii=2;ii<5;ii++)
{
  if(xmin>currentlayer.poly[0].x[ii])
     xmin=currentlayer.poly[0].x[ii];
  if(xmax<currentlayer.poly[0].x[ii])
     xmax=currentlayer.poly[0].x[ii];
  if(ymin>currentlayer.poly[0].y[ii])
     ymin=currentlayer.poly[0].y[ii];
  if(ymax<currentlayer.poly[0].y[ii])
     ymax=currentlayer.poly[0].y[ii];
}

if( (mode==SHOWCUT)&&(objectdeep[currentlayer.num]==0) )
{
  fclose(datacut);
  return;
}
zcut= objectdeep[currentlayer.num];

switch(mode)
{
  case SHOWDXF:
    setcolor(GREEN);
    Rectangledxf(xmax,ymax,xmin,ymin);
  break;
  case SHOWTEST:

    setcolor(GREEN);
    rectangle(xmax,suby-ymax,xmin,suby-ymin);
  break;
}

if( (ofs=fopen(OFFSETNAME,"rt"))==NULL )
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{
 /*closegraph();*/
 printf("can't open offset file for save");
 exit(0);
}

for(yl=ymin;yl<=ymax;yl+=toolr)
{
 switch(mode)
 {
  case SHOWTEST:
            setcolor(GREEN);
            line(xmin,maxy-yl,xmax,maxy-yl);
            setcolor(BLUE);
 }
 /*-------------------------------------*/
 /*                                     */
 /*      FIND  X-CROSS  LINE    */
 /*                                     */
 /*-------------------------------------*/

 ncross=0;
 for(i=0;i<50;i++)
 {
   cross[i].x=0;  cross[i].y=0;
 }
 cross[0].x=xmin; cross[0].y=yl;
 cross[1].x=xmax; cross[1].y=yl;
 ncross=2;
 rewind(ofs);
 do{
  fgets(text,256,ofs);
  if(text[0]=='L')
  {
          getvalue( 2,text,&x1);
          getvalue(19,text,&y1);
          getvalue(36,text,&x2);
          getvalue(53,text,&y2);

          switch(mode)
          {
            case SHOWTEST:
                    setcolor(LIGHTBLUE);
                    line(x1,suby-y1,x2,suby-y2);
                    setcolor(YELLOW);
                    line(x1,suby-y1,x2,suby-y2);
             break;
          }

          if( ((y1>=yl)&&(yl>=y2)) || ((y1<=yl)&&(yl<=y2)) )
          {
            /*                                  */
            /*--- find (xx,yy) cross line(x1,y1,x2,y2)  ---*/
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        /*                                  */
        if( ((y1<=yl)&&(yl<=y2))||((y1>=yl)&&(yl>=y2)) )
        {
          if(y1==y2)
          {
            cross[ncross].x=x1;  cross[ncross].y=y1;
            ncross++;
            cross[ncross].x=x2;  cross[ncross].y=y2;
            ncross++;
          }else
          {
            if(x1==x2)
            {
                    xx=x1;
                    yy=yl;
                    cross[ncross].x=xx;  cross[ncross].y=yy;
                    ncross++;
            }else{
                    m=(y2-y1)/(x2-x1);
                    c=y1-m*x1;
                    yy=yl;
                    xx=(yy-c)/m;
                    cross[ncross].x=xx;  cross[ncross].y=yy;
                    ncross++;
            } /*- end if -*/
          } /*- end if -*/
        } /*- end if -*/
      }
}/*- end if L -*/

if(text[0]=='C')
{
        getvalue( 2,text,&x);
        getvalue(19,text,&y);
        getvalue(36,text,&r);

        switch(mode)
        {
          case SHOWTEST:
                    setcolor(LIGHTBLUE);
                    circle(x,suby-y,r);
                    setcolor(YELLOW);
                    circle(x,suby-y,r);
          break;
        }

        y1=y-r-toolr;  y2=y+r+toolr;

        if( (y1<=yl)&&(yl<=y2) )
        {
          c=r+toolr;
          m=yl-y;
          m=floor(fabs(c*c-m*m)*1.0e12)/1.0e12;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            xx=sqrt(m);

            switch(mode)
            {
              case SHOWTEST:

                        setcolor(LIGHTRED);
                        circle(x+xx,suby-yl,toolr);
                        circle(x-xx,suby-yl,toolr);
              break;
            }
            cross[ncross].x=x+xx;  cross[ncross].y=yl;
            ncross++;
            cross[ncross].x=x-xx;  cross[ncross].y=yl;
            ncross++;
           }
 }
 /*-- ------------------------------------------*/
}while( !feof(ofs) );

/*-------------------------------------------*/
/*                                           */
/*          S O R T   X   C R O S S          */
/*                                           */
/*-------------------------------------------*/
ii=ncross;
for(i=0;i<ncross;i++)
{
          if( (cross[i].x!=0.0)&&(cross[i].y!=0.0) )
          {
            for(j=0;j<ncross;j++)
            {
              if((i!=j)&&(fabs(cross[i].x-cross[j].x)<1.0e-10)&&
                 (fabs(cross[i].y-cross[j].y)<1.0e-10)   )
              {
                        cross[j].x=0.0;cross[j].y=0.0;
                        ii=ii-1;
              } /*- End if ((i    -*/
            } /*- End for j     -*/
          } /*- End if (cross -*/
 } /*- End for i     -*/
 ncross=ii;
 if(ncross>1)
 {
            for(i=ncross-1;i>0;i--)
            {
              for(j=0;j<i;j++)
              {
                if(cross[j].x<cross[j+1].x)
                {
                        xx=cross[j].x;cross[j].x=cross[j+1].x;cross[j+1].x=xx;
                        yy=cross[j].y;cross[j].y=cross[j+1].y;cross[j+1].y=yy;
                }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            }
          }
          /*-----------------------*/
          /*------ draw line -------*/
          /*-----------------------*/
          switch(mode)
          {
            case SHOWCUT:
                    for(i=1;i<ncross;i+=2)
                    {
                     fprintf(datacut,"L %016.12f %016.12f %016.12f %016.12f %016.12f %016.12f \n",
                            cross[i].x  -dxfx1 ,cross[i].y   -dxfy1, zcut,
                            cross[i-1].x-dxfx1 ,cross[i-1].y -dxfy1, zcut);
                    }
            case SHOWTEST:
                    for(i=0;i<ncross;i+=2)
                    {
                      setcolor(MAGENTA);
                      line( cross[i].x   ,suby-cross[i].y,
                            cross[i+1].x ,suby-cross[i+1].y);
                      setcolor(LIGHTBLUE);
                      circle(cross[i].x  ,suby-cross[i].y  ,toolr);
                      circle(cross[i+1].x,suby-cross[i+1].y,toolr);
                    }
            break;
            case SHOWDXF:
                    setcolor(fillcol);
                    for(i=0;i<ncross;i+=2)
                    {
                     Linedxf( cross[i].x   ,cross[i].y,
                              cross[i+1].x ,cross[i+1].y);
                    }
            break;
          }
    } /*-- end if (ncross>1) --*/
  } /*-- end for yl --*/

  fclose(ofs);
  fclose(datacut);
  toolr=savetools;
}

void getvalue(int i,char *text,double *v)
{
  char temp[30];
  int j=0;

  while(text[i]>32)
  {
    temp[j]=text[i];
    j++;
    i++;
  }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  temp[j]=0;
  *v=atof(temp);
}

int read_file_dxf(void)
{

  if( open_dxf_file(dxffile)==YES)
  {
    if(search_entities()==YES)
    {

      toolr=2.0;
      init_dxf_data();
      read_dxf_data();
      calc_dxf_data();
      calc_dxf_max_data();
      calc_scale_dxf();
      // draw_dxf_data();
    }
    fclose(datadxf);
    return(YES);
  }else
  return(NO);
}

int show_view_dxf_on_screen(void)
{

  if(read_file_dxf()==YES)
  {

      maxx=getmaxx();
      maxy=getmaxy();
      setviewport(0,0,maxx,maxy,1);
      clearviewport();
      frame();
      Menubar(1);

      show_scale_dxf_onscreen();
      draw_dxf_fill(SHOWDXF);

      draw_deep_dxf();

    return(YES);
  }else
    return(NO);
}


/*---------------------- Show dxf ----------------*/

void calc_scale_dxf(void)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{
// char text[255];

  dxfx1=currentlayer.xmin[0];
  dxfx2=currentlayer.xmax[0];

  dxfy1=currentlayer.ymin[0];
  dxfy2=currentlayer.ymax[0];

  dxfy=dxfy2-dxfy1;
  dxfx=dxfx2-dxfx1;

  dxfpx = 20.0;
  dxfpy = 20.0;


  if(unit==INCH)
  {
     dispobjx=dxfx/scaleunit;
     dispobjy=dxfy/scaleunit;
  }else
  {
     dispobjx=dxfx;
     dispobjy=dxfy;
  }
}

void show_scale_dxf_onscreen(void)
{
  maxx=getmaxx();
  maxy=getmaxy();
  setviewport(1,MENU_H+2+1,maxx/2-1,maxy/2-1,1);
  clearviewport();

  maxx=maxx/2-1 ;
  maxy=maxy/2 - MENU_H-2-1 ;
  maxy = maxy-20;

  scaledxfx = (maxx - 2*20)/(dxfx);
  scaledxfy = (maxy - 2*20)/(dxfy);


  /*-------------------------------------------*/
  /*              set output view              */
  /*-------------------------------------------*/
  ;
  sprintf(text,"Dxf = (%0.3f,%0.3f) ",dispobjx,dispobjy);
  if(unit==INCH)
    strcat(text," In.");
  else
    strcat(text," mm.");
  setcolor(WHITE);
  outtextxy(20,maxy,text);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
}

void Circledxf(double x,double y,double r)
{
  x = x - dxfx1;  y = y - dxfy1;

  x = x * scaledxfx; y = y * scaledxfy;

  x = x + dxfpx;     y = y + dxfpy;

  y = maxy - y;

  r=r*1.0;
  // circle( x,y,r);
}

void Linedxf(double x1,double y1,double x2,double y2)
{
  x1 = x1 - dxfx1;  y1 = y1 - dxfy1;
  x2 = x2 - dxfx1;  y2 = y2 - dxfy1;

  x1 = x1 * scaledxfx; y1 = y1 * scaledxfy;
  x2 = x2 * scaledxfx; y2 = y2 * scaledxfy;

  x1 = x1 + dxfpx;     y1 = y1 + dxfpy;
  x2 = x2 + dxfpx;     y2 = y2 + dxfpy;

  y1 = maxy - y1;
  y2 = maxy - y2;

  line(  x1, y1 , x2, y2 );
 }

 void Rectangledxf(double x1,double y1,double x2,double y2)
 {

   x1 = x1 - dxfx1;  y1 = y1 - dxfy1;
   x2 = x2 - dxfx1;  y2 = y2 - dxfy1;

   x1 = x1 * scaledxfx; y1 = y1 * scaledxfy;
   x2 = x2 * scaledxfx; y2 = y2 * scaledxfy;

   x1 = x1 + dxfpx;     y1 = y1 + dxfpy;
   x2 = x2 + dxfpx;     y2 = y2 + dxfpy;

   y1 = maxy - y1;
   y2 = maxy - y2;


    rectangle( x1,y1,x2,y2);
  }

   void draw_deep_dxf(void)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{
  // int i;

  maxx = getmaxx();
  maxy = getmaxy();

  setviewport(maxx/2+1 , MENU_H+2+1,maxx-1,maxy/2-1,1);
  clearviewport();

  maxx=maxx/2-1 ;
  maxy=maxy/2 - MENU_H-2-1 ;

  strcpy(text," เครื่องมือ  ขนาด  ความลึก");
  if(unit==INCH)
    strcat(text,"(นิ้ว)");
  else
    strcat(text,"(มิล.)");

  drawtext(40,10,text,WHITE);

}

void frame(void)
{
  int col=getcolor();
  char text[255];
  char unittext[10],speedtext[15];
  double posx,posy,posz;
  int i,j;

  setcolor(WHITE);

  rectangle(0,0,maxx,maxy);
  rectangle(0,maxy/2,maxx,maxy/2);
  rectangle(maxx/2,0,maxx/2,maxy-64);

  line(0,maxy-16,maxx,maxy-16);
  line(0,maxy-32,maxx,maxy-32);
  line(0,maxy-48,maxx,maxy-48);
  line(0,maxy-64,maxx,maxy-64);

  setfillstyle(SOLID_FILL,BLACK);
  bar(180,maxy-63,500,maxy-49);
  bar(180,maxy-47,500,maxy-33);

  axis(50.0);

  block3d(0.0,0.0,0.0,objectwidth,objectlength,-objectheight,COL_OBJECT);
  draw_dxf_3d();

  draw_head();
  /*-------------------------------*/
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
*/
settextjustify(0,1);
setcolor(WHITE);

if(unit==INCH)
{
  posx=abs_pos_x/scaleunit;
  posy=abs_pos_y/scaleunit;
  posz=abs_pos_z/scaleunit;
  strcpy(unittext,"inch");
  strcpy(speedtext,"inch/min");
}else
{
  posx=abs_pos_x;
  posy=abs_pos_y;
  posz=abs_pos_z;
  strcpy(unittext," mm.");
  strcpy(speedtext," mm./sec.");
}*/

sprintf(text,"Absolute Position X=%8.3f Y=%8.3f Z=%8.3f %s",posx,posy,posz,unittext);
//  strcat(text,unittext);
outtextxy(20,maxy-56,text);

if(unit==INCH)
{
  posx=rel_pos_x/scaleunit;
  posy=rel_pos_y/scaleunit;
  posz=rel_pos_z/scaleunit;
}else
{
  posx=rel_pos_x;
  posy=rel_pos_y;
  posz=rel_pos_z;
}
sprintf(text,"Relative Position X=%8.3f Y=%8.3f Z=%8.3f %s",posx,posy,posz,unittext);
outtextxy(20,maxy-40,text);

sprintf(text,"Feed rate    : %.4f %s",feedrate_value,speedtext);
outtextxy(maxx/2+100,maxy-56,text);
sprintf(text,"Spindle rate : %.4f rpm.",spindle_value);
outtextxy(maxx/2+100,maxy-40,text);
sprintf(text,"Jog Speed    : %.4f %s",jog_value,speedtext);
outtextxy(maxx/2+100,maxy-24,text);

/*-------------------------------------------------*/
line(  maxx/3,maxy-16,  maxx/3,maxy);
line(2*maxx/3,maxy-16,2*maxx/3,maxy);

i=strlen(prjfile);
if(i>0)
{
  while(prjfile[i]!='\\') i--;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  i++;
  j=0;  while(prjfile[i]!=0) {   tmpname[j]=prjfile[i];   j++;i++;  }
  tmpname[j]=0;
 }else strcpy(tmpname,"");
 strcpy(text,"Prj : ");strcat(text,tmpname);
 outtextxy(20,maxy-8,text);

 i=strlen(dxffile);
 if(i>0)
 {
  while(dxffile[i]!='\\')i--;i++;j=0;
  while(dxffile[i]!=0)
  {
    tmpname[j]=dxffile[i];
    j++;i++;
  }
  tmpname[j]=0;
 }else strcpy(tmpname,"");

 strcpy(text,"Dxf : ");strcat(text,tmpname);
 outtextxy(maxx/3+20,maxy-8,text);

 strcpy(text,"Mch : ");strcat(text,mchname); // rew
 strcat(text,".cod");
 outtextxy(2*maxx/3+20,maxy-8,text);

 if(unit==INCH)
 {
   dispobjx=objectwidth/scaleunit;
   dispobjy=objectlength/scaleunit;
   dispobjz=objectheight/scaleunit;
 }
 else
 {
   dispobjx=objectwidth;
   dispobjy=objectlength;
   dispobjz=objectheight;
 }
 /*-------------------------------------------*/
 sprintf(text,"Object (%0.4f,%0.4f,%0.4f) ",
                                    dispobjx,dispobjy,dispobjz);
 if(unit==INCH)
   strcat(text," inch");
 else
   strcat(text," mm.");

 outtextxy(20,maxy-24,text);
 setcolor(col);
 settextjustify(0,0);
}

void compile_code(void)
{
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
double x1,y1,z1,x2,y2,z2;
double x,y,z,r;
int col;
int linenum=10;
//int status;

//a=b=c=-0.5;
if( (cutfile=fopen(CUTNAME,"rt"))==NULL )
{
  printf("Can't open file name.");
  exit(0);
}

strcpy(tmpname,path_data_file);
strcat(tmpname,mchname);
strcat(tmpname,".cod");

if( (mchfile=fopen(tmpname,"wt"))==NULL )
{
  printf("Can't open file name.");
  exit(0);
}
if( (isofile=fopen(ISONAME,"wt"))==NULL )
{
  printf("Can't open file name.");
  exit(0);
}

fprintf(mchfile,"N%d G00 X0 Y0 Z10\n",linenum);
linenum+=linestep;
fprintf(isofile,"block(%.4f,%.4f,%.4f)\n",objectx,objecty,objectz);

fprintf(mchfile,"N%d G17 T1\n",linenum);
linenum+=linestep;

rewind(cutfile);
col=0;
do{
  fgets(text,256,cutfile);
  if(text[0]=='C')
  {
    getvalue( 2,text,&x);
    getvalue(19,text,&y);
    getvalue(36,text,&z);
    getvalue(53,text,&r);
    x1=x-r;y1=y;
    if(col>0)
    {
          //xline(a,b,0.5);
          fprintf(mchfile,"N%d G01 Z%.3f F%d\n",linenum,5.0,fastspeed);
          fprintf(isofile,"line(x,y,%.4f)\n",5.0);
          linenum+=linestep;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        //xline(x1,y1,0.5);
        fprintf(mchfile,"N%d G00 X%.3f Y%.3f \n",linenum,x1,y1);
        fprintf(isofile,"line(%.4f,%.4f,Z)\n",x1,y1);
        linenum+=linestep;

        //xline(x1,y1,-z);
        fprintf(mchfile,"N%d G01 Z%.3f F%d\n",linenum,-z,slowspeed);
        fprintf(isofile,"line(x,y,%.4f)\n",-z);
        linenum+=linestep;

        //xcircle(x1,y1,r);
        fprintf(mchfile,"N%d G02 X%.3f Y%.3f I%.3f J%.3f F%d\n",
                                linenum,x1,y1,r,0.0,curvespeed);
        fprintf(isofile,"circle(%.4f,%.4f,%.4f,%.4f)\n",x,y,z,r);
        linenum+=linestep;

}else
{
        //xline(x1,y1,0.5);
        fprintf(mchfile,"N%d G00 X%.3f Y%.3f Z%.3f\n",linenum,x1,y1,5.0);
        fprintf(isofile,"line(%.4f,%.4f,%.4f)\n",x1,y1,5.0);
        linenum+=linestep;

        //xline(x1,y1,-z);
        fprintf(mchfile,"N%d G01 X%.3f Y%.3f Z%.3f F%d\n",linenum,x1,y1,-z,slowspeed);
        fprintf(isofile,"line(%.4f,%.4f,%.4f)\n",x1,y1,-z);
        linenum+=linestep;

        //xcircle(x1,y1,r);
        fprintf(mchfile,"N%d G02 X%.3f Y%.3f I%.3f J%.3f F%d\n",linenum,x1,y1,r,0.0,curvespeed);
        fprintf(isofile,"circle(%.4f,%.4f,%.4f,%.4f)\n",x,y,z,r);
        linenum+=linestep;
 }
        //a=x1;  b=y1;
}
if(text[0]=='L')
{
  getvalue( 2,text,&x1);
  getvalue(19,text,&y1);
  getvalue(36,text,&z1);
  getvalue(53,text,&x2);
  getvalue(70,text,&y2);
  getvalue(87,text,&z2);


  if(col>0)
  {
        //xline(a,b,0.5);
        fprintf(mchfile,"N%d G01 Z%.3f F%d\n",linenum,5.0,fastspeed);
        fprintf(isofile,"line(x,y,%.4f)\n",5.0);
        linenum+=linestep;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            //xline(x1,y1,0.5);
            fprintf(mchfile,"N%d G00 X%.3f Y%.3f \n",linenum,x1,y1);
            fprintf(isofile,"line(%.4f,%.4f,z)\n",x1,y1);
            linenum+=linestep;


            //xline(x1,y1,-z1);
            fprintf(mchfile,"N%d G01 Z%.3f F%d\n",linenum,-z1,slowspeed);
            fprintf(isofile,"line(x,y,%.4f)\n",-z1);
            linenum+=linestep;

    }else
    {
            col++;
            //xline(x1,y1,0.5);

            fprintf(mchfile,"N%d G00 X%.3f Y%.3f Z%.3f\n",linenum,x1,y1,5.0);
            fprintf(isofile,"line(%.4f,%.4f,%.4f)\n",x1,y1,5.0);
            linenum+=linestep;

            //xline(x1,y1,-z1);
            fprintf(mchfile,"N%d G01 Z%.3f F%d\n",linenum,-z1,slowspeed);
            fprintf(isofile,"line(x,y,%.4f)\n",-z1);
            linenum+=linestep;
    }
    setcolor(col);
    //xline(x2,y2,-z2);
    fprintf(mchfile,"N%d G01 X%.3f Y%.3f F%d\n",linenum,x2,y2,slowspeed);
    fprintf(isofile,"line(%.4f,%.4f,z)\n",x2,y2);
    linenum+=linestep;
    //a=x2;b=y2;c=z2;
    col++;
    if(col==16)col=1;
  }

 } while(!feof(cutfile));

 //xline(a,b,0.5);
 fprintf(mchfile,"N%d G01 Z%.3f \n",linenum,0.5);
 linenum+=linestep;

 fprintf(mchfile,"M02");
 linenum+=linestep;
 fprintf(isofile,"end");
 //getch();
 fclose(cutfile);
 fclose(mchfile);
 fclose(isofile);
}

void Moveline(x1,y1,z1,x2,y2,z2)
double x1,y1,z1,x2,y2,z2;
{
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
 int col;

 CUT=ON;
 if(x1==x2)
   col=COL_CUT_VER;
 else if(y1==y2)
   col=COL_CUT_HOR;
 else
   col=COL_CUT_NORM;

 line3d(x1,y1,z1,x2,y2,z2,col);
 CUT=OFF;
 toolx=x2;
 tooly=y2;
 toolz=z2;
}


double get_pos(int axis)
{
 unsigned p1,p2,pos;
 unsigned port;

 if(axis==AXIS_X) port=HCTL_X;
 if(axis==AXIS_Y) port=HCTL_Y;
 if(axis==AXIS_Z) port=HCTL_Z;

 outportb(port,REG_POS);
 p1=inportb(port);
 outportb(port,REG_POS+1);
 p2=inportb(port);

 pos=(p1<8)|p2;
 return( (double)(pos) );
}

void MoveMachineTo(double mx,double my,double mz,double feed)
{
  double mchx,mchy,mchz;

  mchx=get_pos(AXIS_X);
  mchy=get_pos(AXIS_Y);
  mchz=get_pos(AXIS_Z);

  mx=mx*facmove;
  my=my*facmove;
  mz=mz*facmove;

  if(move==MOVE_INC)
  {
    mx=mchx+mx;
    my=mchy+my;
    mz=mchz+mz;
```

เอกสารนี้มีไว้สำหรับการที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  }
  MoveMachine(mchx,mchy,mchx,mx,my,mz,feed);
}

void set_pos(int axis,double ps)
{
  unsigned int pos=(unsigned)(ps);
  unsigned char p1,p2;
  unsigned port;

  if(axis==AXIS_X) port=HCTL_X;
  if(axis==AXIS_Y) port=HCTL_Y;
  if(axis==AXIS_Z) port=HCTL_Z;


  p1=pos&0x0f;
  p2=(pos&0xf0)>8;

  outportb(port,REG_POS);
  outportb(port,p1);
  outportb(port,REG_POS+1);
  outportb(port,p2);
}

void set_vel(int axis,double vl)
{
  unsigned int vel=(unsigned)(vl);
  unsigned char v1,v2;
  unsigned port;

  if(axis==AXIS_X) port=HCTL_X;
  if(axis==AXIS_Y) port=HCTL_Y;
  if(axis==AXIS_Z) port=HCTL_Z;


  v1=vel&0x0f;
  v2=(vel&0xf0)>8;

  outportb(port,REG_VEL);
  outportb(port,v1);
  outportb(port,REG_VEL+1);
  outportb(port,v2);
}

 void set_move(int move)
 {
  if(move==ON)
  {
   /*- set all axis move -*/
  } else
  {
   /*- set all axis stop -*/
  }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
}

void MoveMachine(double x1,double y1,double z1,double x2,double y2,double z2,double feedrate)
{
  double r;
  double dx,dy,dz;
  double vx,vy,vz;

  dx=x2-x1;
  dy=y2-y1;
  dz=z2-z1;

  r=sqrt(dx*dx + dy*dy + dz*dz);

  vx=feedrate*(dx/r);
  vy=feedrate*(dy/r);
  vz=feedrate*(dz/r);

  if (vx<0) vx=-vx;
  if (vy<0) vy=-vy;
  if (vz<0) vz=-vz;

  set_move(OFF);
  set_pos(AXIS_X,x2);
  set_vel(AXIS_X,vx);
  set_pos(AXIS_Y,y2);
  set_vel(AXIS_Y,vy);
  set_pos(AXIS_Z,z2);
  set_vel(AXIS_Z,vz);
  set_move(ON);
}

void MoveLineTo(double x,double y,double z)
{
   Moveline(toolx,tooly,toolz,x,y,z);
}

void MoveLineInc(double x,double y,double z)
{
   Moveline(toolx,tooly,toolz,toolx+x,tooly+y,toolz+z);
}

void HomeX(void)
{
  Moveline(toolx,tooly,toolz,0.0,tooly,toolz);
}

void HomeY(void)
{
  Moveline(toolx,tooly,toolz,toolx,0.0,toolz);
}

  void HomeZ(void)
```

เอกสารนี้มีไว้สำหรับการที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{
  Moveline(toolx,tooly,toolz,toolx,tooly,0.0);
}

void MoveArcXY(dir,cx,cy,r,p,startang,endang)
int dir;
double cx,cy,r,p,startang,endang;
{
  arc3dXY(dir,cx,cy,p,r,startang,endang,COLORPATH);
}

void MoveArcXZ(dir,cx,cz,r,p,startang,endang)
int dir;
double cx,cz,r,p,startang,endang;
{
  arc3dXZ(dir,cx,p,cz,r,startang,endang,COLORPATH);
}

void MoveArcYZ(dir,cy,cz,r,p,startang,endang)
int dir;
double cy,cz,r,p,startang,endang;
{
  arc3dYZ(dir,p,cy,cz,r,startang,endang,COLORPATH);
}

void setColor256(void)
{ int i;

  for(i=WHITE+1;i< 64;i++) setrgbpalette(i, 63-i,63-i,63-i);

  for(i=0;i<64;i++) setrgbpalette(COLRED  +i, 63  , 63-i, 63-i);
  for(i=0;i<64;i++) setrgbpalette(COLGREEN+i, 0   ,   i, 0   );
  for(i=0;i<64;i++) setrgbpalette(COLBLUE +i, 63-i, 63-i, 63  );
}

void showColor256(void)
{
   int i;

   for(i=0;i<256;i++)
   {
     setcolor(i);
     line(20+i,100,20+i,200);
   }
   getch();
}

void TestObject(void)
{
  unsigned int x,y,z;
  unsigned i,j,k;
  double deg1,deg2;
  double t1,t2;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  double dx,dy;

  x=ReadXObject();
  y=ReadYObject();
  z=ReadZObject();

  dx=  360.0/x;
  dy=  360.0/y;

  deg1=0.0;
  deg2=0.0;

  for(i=0;i<x;i++,deg1+=dx)
  {
    t1=sin(deg1*M_PI/180.0);
    for(j=0;j<y;j++,deg2+=dy)
    {
      t2=sin(deg2*M_PI/180.0);
      if( !(((i<20) /*||(i>x-20)*/) || ((j<20) /*||(j>y-20)*/ )) )
      {
            k=(int)(t1*40) + (int)(t2*55) + (int)(z/2);
            if(k<4)k=5;
            if(k>z)k=z-20;
      } else k=z;
      k=WriteObject(i,j,k);
    }
  }
  outtextxy(300,maxy-25,"Press any key to continue.");
  cleardevice();
}

void ViewSolidObject(void)
{
  unsigned int x,y,z;
  unsigned int i,j;
  double xx,yy,zz;
  double dx,dy,dz;
  double cx,cy,//,cz;

  x=ReadXObject();
  y=ReadYObject();
  z=ReadZObject();

  dx=64.0/x;
  dy=64.0/y;
  dz=64.0/z;

  if(view5==OFF)
    view1=ON;
  for(i=0,cx=0;i<x;i++,cx+=dx)
  {
    xx=(double)(i);
    for(j=0,cy=0;j<y;j++,cy+=dy)
```

เอกสารนี้จัดเตรียมไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    {
      yy=(double)(j);
      zz=ReadObject(i,j);
    "
      if(view5==OFF)
      {
            view3=view4=ON;
            view2=ON;
      }
      line3df(xx,yy,  -objectheight ,xx,yy,zz-objectheight,(int)(COLRED + cy));
      if(view5==OFF)
      {
            view4=view2=view3=ON;
      }
      plot3df(xx,yy,zz-objectheight ,(int)(COLGREEN + zz*dz-1));
    }
    if(view5==OFF)
    {
      view2=view4=ON;
      view3=ON;
    }
    line3df(xx,yy,-objectheight,xx,yy,zz-objectheight,(int)(COLBLUE + cx));
  }
  if(view5==OFF)
  {
    view1=view2=view3=view4=ON;
  }
}

void ViewSolidObject64(void)
{
  unsigned int x,y,z;
  double i,j;
  unsigned int a,b;
  double c;
  double fx,fy,fz;
  double dx,dy,dz,dmax;
  double cx,cy,cz;
  int savemodeview;

  savemodeview=view4;
  view4=OFF;

  x=ReadXObject();
  y=ReadYObject();
  z=ReadZObject();

  dmax=x;
  if(y>dmax)dmax=y;
  if(z>dmax)dmax=z;

  dx=x*SCALEOBJ/dmax;
  dy=y*SCALEOBJ/dmax;
```

เอกสารนี้มีลิขสิทธิ์ สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
dz=z*SCALEOBJ/dmax;

cx=64.0/dx;  cy=64.0/dy;  cz=64.0/dz;
fx=x/dx;     fy=y/dy;     fz=dz/z;

for(i=0.0;i<dx;i+=1)
{
  a=(int)(i*fx);
  for(j=0.0;j<dy;j+=1)
  {
    b=(int)(j*fy);
    c=ReadObject(a,b);
    c=c*fz;
    line3df(i,j,0,i,j,c-fz,(int)(COLRED + (int)(j*cy)));
    plot3df(i,j,c,(int)(COLGREEN + (int)(c*cz)-1));
    putpixel(dx-i+20,j+20,(int)(COLGREEN+c*cz-1));
  }
  line3df(i,j,0, i,j,c,(int)(COLBLUE +(int)(i*cx)));
}
outtextxy(100,maxy-20,"Press any key to continue.");
view4=savemodeview;
}

double readnum(char text[],int *i)
{
  int j=0;
  char txtnum[40];

  while( (text[*i]!=' ')&&(text[*i]!='\n') )
  {
    txtnum[j]=text[*i];
    j++; *i+=1;
  }
  txtnum[j]=0;
  *i+=1;
  return(atof(txtnum) );
}

void do_gcode_00(int doFlag,double x,double y,double z,int moveFlag)
{
  double mx,my,mz;

  mx=toolx;
  my=tooly;
  mz=toolz;

  if( (moveFlag&MOVE_X)!=0 ) mx=x;
  if( (moveFlag&MOVE_Y)!=0 ) my=y;
  if( (moveFlag&MOVE_Z)!=0 ) mz=z;

  MoveLineTo(mx,my,mz);
  if(doFlag==DO_CYCLE)
    MoveMachineTo(mx,my,mz,FEEDMAX);
```

เอกสารนี้มีลิขสิทธิ์ สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
}


void do_gcode_01(int doFlag,double x,double y,double z,int moveFlag,double feedrate)
{
  double mx,my,mz;

  mx=toolx;
  my=tooly;
  mz=toolz;

  if( (moveFlag&MOVE_X)!=0 ) mx=x;
  if( (moveFlag&MOVE_Y)!=0 ) my=y;
  if( (moveFlag&MOVE_Z)!=0 ) mz=z;

  MoveLineTo(mx,my,mz);
  if(doFlag==DO_CYCLE)
    MoveMachineTo(mx,my,mz,feedrate);
}

void do_gcode_02(int doFlag,double numX,double numY,double numZ,double numI,double numJ,double numK,int moveFlag,double feedrate)
{
  double r;
  double x,y,z,sx,sy,sz,ex,ey,ez;
  double cx,cy,cz;
  double start,end;

  sx=toolx;   sy=tooly;   sz=toolz;
  ex=numX;    ey=numY;    ez=numZ;
  x=numI;     y=numJ;     z=numK;
  cx=sx+x;    cy=sy+y;    cz=sz+z;
  r=sqrt(x*x + y*y + z*z);

  if(moveFlag==MOVE_XY)
  {
    start = atan2(sy-cy,sx-cx);
    end   = atan2(ey-cy,ex-cx);
    start=start*180.0/M_PI;
    end=end*180.0/M_PI;
    MoveArcXY(CW_DIR,cx,cy,r,toolz,start,end);
    MoveLineTo(ex,ey,toolz);
    if(doFlag==DO_CYCLE)
      MoveMachineTo(ex,ey,toolz,feedrate);
  }
  if(moveFlag==MOVE_YZ)
  {
    start = atan2(sz-cz,sy-cy);
    end   = atan2(ez-cz,ey-cy);
    start=start*180.0/M_PI;
    end=end*180.0/M_PI;
    MoveArcYZ(CW_DIR,cy,cz,r,toolx,start,end);
    MoveLineTo(toolx,ey,ez);
    if(doFlag==DO_CYCLE)
```

เอกสารนี้มีลิขสิทธิ์สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    MoveMachineTo(toolx,ey,ez,feedrate);
  }
  if(moveFlag==MOVE_XZ)
  {
   start = atan2(sz-cz,sx-cx);
   end   = atan2(ez-cz,ex-cx);
   start=start*180.0/M_PI;
   end=end*180.0/M_PI;
   MoveArcXZ(CW_DIR,cx,cz,r,tooly,start,end);
   MoveLineTo(ex,tooly,ez);
   if(doFlag==DO_CYCLE)
    MoveMachineTo(ex,tooly,ez,feedrate);
  }
}

void do_gcode_03(int doFlag,double numX,double numY,double numZ,double numI,double numJ,double numK,int moveFlag,double feedrate)
{
  double r;
  double x,y,z,sx,sy,sz,ex,ey,ez;
  double cx,cy,cz;
  double start,end;

  sx=toolx;   sy=tooly;   sz=toolz;
  ex=numX;    ey=numY;    ez=numZ;
  x=numI;     y=numJ;     z=numK;
  cx=sx+x;    cy=sy+y;    cz=sz+z;
  r=sqrt(x*x + y*y + z*z);

  if(moveFlag==MOVE_XY)
  {
   start = atan2(sy-cy,sx-cx);
   end   = atan2(ey-cy,ex-cx);
   start=start*180.0/M_PI;
   end=end*180.0/M_PI;
   MoveArcXY(CCW_DIR,cx,cy,r,toolz,start,end);
   MoveLineTo(ex,ey,toolz);
   if(doFlag==DO_CYCLE)
    MoveMachineTo(ex,ey,toolz,feedrate);
  }
  if(moveFlag==MOVE_YZ)
  {
   start = atan2(sz-cz,sy-cy);
   end   = atan2(ez-cz,ey-cy);
   start=start*180.0/M_PI;
   end=end*180.0/M_PI;
   MoveArcYZ(CCW_DIR,cy,cz,r,toolx,start,end);
   MoveLineTo(toolx,ey,ez);
   if(doFlag==DO_CYCLE)
    MoveMachineTo(toolx,ey,ez,feedrate);
  }
  if(moveFlag==MOVE_XZ)
  {
   start = atan2(sz-cz,sx-cx);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    end   = atan2(ez-cz,ex-cx);
    start=start*180.0/M_PI;
    end=end*180.0/M_PI;
    MoveArcXZ(CCW_DIR,cx,cz,r,tooly,start,end);
    MoveLineTo(ex,tooly,ez);
    if(doFlag==DO_CYCLE)
      MoveMachineTo(ex,ey,toolz,feedrate);
  }
}

void do_gcode_04(double numP)
{
  delay(numP*1000);
}

void do_gcode_20(void)
{
  unit=INCH;
  facmove=FACINCH;
}

void do_gcode_21(void)
{
  unit=CENT;
  facmove=FACCENT;
}

void do_gcode_90(void)
{
  move=MOVE_ABS;
}

void do_gcode_91(void)
{
  move=MOVE_INC;
}

void do_mcode_02(void)      /*  stop program  */
{
  stop_all_action();
  move_home();
  /*
  int i;

  for (i=0;i<=2;i++)
  {
            change_axis(i);
            axis_on(0);
            mtr_on(0);
            set_output_dac16(0);
            set_prfl_trap(0);
            set_acc(acc);
            set_vel(0L);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        set_fltr_pid(0);
        set_kp(10L);
        set_ki(10L);
        set_kd(10L);
        set_capt_home(0);
        synch_prfl(0);
        update(0);
    }
    set_4(0);
    set_mtr_cmd(0);
    update(0);

}

void do_mcode_03(void)      /* spindle on cw  */
{
    unsigned spindle;

    if(FlagAction==DO_SIM)
        return;

    spindle=(int)((float)speed*32767/1500);
    if (spindle<0) spindle=-spindle;
    set_mtr_cmd(spindle);
}

void do_mcode_04(void)      /* spindle on ccw */
{
    unsigned spindle;

    if(FlagAction==DO_SIM)
        return;

    spindle=(int)((float)speed*32767/1500);
    if (spindle<0) spindle=-spindle;
    set_mtr_cmd(spindle);
}

void do_mcode_05(void)      /* spindle off    */
{
    if(FlagAction==DO_SIM)
        return;

    change_axis(3);
    axis_on(0);
    mtr_on(0);
    set_output_dac16(0);
    set_prfl_vel(0);
    set_acc(acc);
    set_vel(0L);
    set_fltr_pid(0);
    set_kp(10L);
    set_ki(10L);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  set_kd(10L);
  set_capt_home(0);
  synch_prfl(0);
  update(0);
}

/*------------------- m a i n ---------------------*/
void initgph(void)
{
  int i,j;

  detectgraph(&i,&j);
  initgraph(&i,&j,"");
}

/*-------------------------------------------------*/
void InitBuffer(void)
{
  int i,j;

  for(i=0;i<BUFMAX;i++)
    for(j=0;j<BUFMAX;j++)
      buffer[i][j]=0;

}

void simulate(void)
{
  double numX,numY,numZ,numI,numJ,numK,numN,numR,numP,numS,numF;
  int moveX,moveY,moveZ,moveFlag;
  int numG,numM;
  int i,//,j;
  int quit=OFF;
  int ch,chf;

  toolx=tooly=toolz=200.0;
  InitObject(objectwidth,objectlength,objectheight);

  strcpy(tmpname,path_data_file);
  strcat(tmpname,mchname);
  strcat(tmpname,".cod");

  if( (code=fopen(tmpname,"rt"))==NULL )
  {
    closegraph();
    printf("\nCan't open file name.");
    printf("\nExit Program.");
    getch();
    exit(0);
  }

  do{
    i=0;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
numX=numY=numZ=numI=numJ=numK=numN=numR=0.0;
numP=numS=numF=-1.0;
numG=numM=-1;
moveX=moveY=moveZ=OFF;

fgets(text,255,code);

dispblank_xy_text (10,25*20,60,BLACK);
drawtextbackground(10,25*20,text,LIGHTRED,BLUE,DARKGRAY);

i=0;
while( ((ch=text[i])!='\n')&&(i<strlen(text)) )
{
 i++;
 switch(ch)
 {
        case 'n':case 'N':
          numN=readnum(text,&i);
        break;
        case 'g':case 'G':
          numG=(int)(readnum(text,&i));
        break;
        case 'm':case 'M':
          numM=(int)(readnum(text,&i));
        break;
        case 'x':case 'X':
          numX=readnum(text,&i);
          moveX=ON;
        break;
        case 'y':case 'Y':
          numY=readnum(text,&i);
          moveY=ON;
        break;
        case 'z':case 'Z':
          numZ=readnum(text,&i);
          moveZ=ON;
        break;
        case 'i':case 'I':
          numI=readnum(text,&i);
        break;
        case 'j':case 'J':
          numJ=readnum(text,&i);
        break;
        case 'k':case 'K':
          numK=readnum(text,&i);
        break;
        case 'r':case 'R':
          numR=readnum(text,&i);
        break;
        case 'p':case 'P':
          numP=readnum(text,&i);
        break;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        case 'f':case 'F':
          numF=readnum(text,&i);
        break;
        case 's':case 'S':
          numS=readnum(text,&i);
        break;
  }
}

if(numG!=-1)
{
  moveFlag=(moveX<<2)|(moveY<<1)|moveZ;
  if(numF!=-1.0)
  {
        feedrate_value=numF;
  } else numF=feedrate_value;
  switch(numG)
  {
        case 00:/*----------------------*/
          do_gcode_00(FlagAction,numX,numY,numZ,moveFlag);
        break;
        case 01:/*----------------------*/
          do_gcode_01(FlagAction,numX,numY,numZ,moveFlag,numF);
        break;
        case 02:/*----------------------*/
          do_gcode_02(FlagAction,numX,numY,numZ,numI,numJ,numK,moveFlag,numF);
        break;
        case 03:/*----------------------*/
          do_gcode_03(FlagAction,numX,numY,numZ,numI,numJ,numK,moveFlag,numF);
        break;
        case 04:/*----------------------*/
          do_gcode_04(numP);
        break;
        case 20: /* unit = INCH */
          do_gcode_20();
        break;
        case 21: /* unit = CENT */
          do_gcode_21();
        break;
        case 90: /* move = ABS  */
          do_gcode_90();
        break;
        case 91: /* move = INC  */
          do_gcode_91();
        break;
  }
}
if(numM!=-1)
{
  switch(numM)
  {
        case 02:
          do_mcode_02();  /* stop program  */
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                break;
                case 03:
                  do_mcode_03();      /* spindle on cw  */
                break;
                case 04:
                  do_mcode_04();      /* spindle on ccw */
                break;
                case 05:
                  do_mcode_05();      /* spindle off    */
                break;
      }
    }
    /*-  chk cycle stop -*/
    if(kbhit())
    {
      ch=getch();chf=0;
      if(ch==0)chf=getch();
    }
    if((ch==ESC)||(ch==ENTER))
      quit=ON;
  } while( (!feof(code)) && (quit!=ON) );
  fclose(code);
}

void getvalue_tools(void)
{
  int i;
  int ch;
  int col;
  double s,h;

  do{
    framebox(30, 6, 80-10, 15, DARKGRAY);
    drawtext((25+10)*8, 8*20-10,"เครื่องมือ     ขนาด     ความยาว",WHITE);

    for(i=0;i<5;i++)
    {
      sprintf(text,"  %d       %6.3f      %6.3f",i+1,toolsize[i],tooldeep[i]);
      drawtext((25+10)*8, (9+i)*20-10,text,WHITE);
    }

    drawtext((25+10)*8, 286,"C-เปลี่ยนค่าเครื่องมือ Esc-กลับเมนูหลัก ",WHITE);
    ch=getch();

    switch(ch)
    {
      case 'c':
      case 'C':
                col=RED; //DARKGRAY;
                dispblank_xy_text((25+10)*8,13*22,32,col);
                drawtext((25+10)*8,13*22,"เบอร์เครื่องมือ : ",WHITE);
                strcpy(text,"");
                getstring(text,40+10,13,5,NUMBER,LIGHTRED);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            i=atoi(text);
            dispblank_xy_text((25+10)*8,13*22,32,col);
            drawtext((25+10)*8,13*22,"ขนาด :          ยาว :",WHITE);
            strcpy(text,"");
            getstring(text,32+10,13,5,NUMBER,LIGHTRED);
            s=atof(text);
            strcpy(text,"");
            getstring(text,47+10,13,5,NUMBER,LIGHTRED);
            h=atof(text);
      break;
    }

  }while(ch!=ESC);
}

void out_position(void)
{
  char text[255];
  double posx,posy,posz;

  setfillstyle(SOLID_FILL,BLACK);
  bar(180,maxy-63,500,maxy-49);
  bar(180,maxy-47,500,maxy-33);


  settextjustify(0,1);
  setcolor(WHITE);

  if(unit==INCH)
  {
    posx=abs_pos_x/scaleunit;
    posy=abs_pos_y/scaleunit;
    posz=abs_pos_z/scaleunit;
  }else
  {
    posx=abs_pos_x;
    posy=abs_pos_y;
    posz=abs_pos_z;
  }

  sprintf(text,"Absolute Position X=%8.3f Y=%8.3f Z=%8.3f",posx,posy,posz);
  if(unit==INCH)
    strcat(text," inch");
  else
    strcat(text," mm.");
  outtextxy(20,maxy-56,text);

  rel_pos_x=abs_pos_x-set_pos_x;
  rel_pos_y=abs_pos_y-set_pos_y;
  rel_pos_z=abs_pos_z-set_pos_z;

  if(unit==INCH)
  {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    posx=rel_pos_x/scaleunit;
    posy=rel_pos_y/scaleunit;
    posz=rel_pos_z/scaleunit;
  }else
  {
    posx=rel_pos_x;
    posy=rel_pos_y;
    posz=rel_pos_z;
  }
  sprintf(text,"Relative Position X=%8.3f Y=%8.3f Z=%8.3f",posx,posy,posz);
  if(unit==INCH)
    strcat(text," inch");
  else
    strcat(text," mm.");
  outtextxy(20,maxy-40,text);

}

void set_feed_rate(void)
{

  framebox(20, 7, 80, 12, DARKGRAY);
  drawtext(44*8, 9*20-10,"ตั้งค่า Feed Rate",WHITE);
  sprintf(text,"ค่า Feed Rate เดิม : %.4f",feedrate_value);
  drawtext(30*8,10*20,"ค่า Feed Rate = ",WHITE);

  strcpy(text,"");getstring(text,46,9,10,NUMBER,LIGHTRED);
  if(strcmp(text,"")!=0)
    feedrate_value=atof(text);
}

void set_spindle_rate(void)
{
  framebox(20, 7, 80, 12, DARKGRAY);
  drawtext(44*8, 9*20-10,"ตั้งค่า Spindle rate",WHITE);
  sprintf(text,"ค่า Spindle Rate เดิม : %.4f",spindle_value);
  drawtext(34*8,10*20,"ค่า Spindle Rate = ",WHITE);

  strcpy(text,"");getstring(text,54,9,10,NUMBER,LIGHTRED);
  if(strcmp(text,"")!=0)
    spindle_value=atof(text);
}

void set_jog_speed(void)
{
  framebox(20, 7, 80, 12, DARKGRAY);
  drawtext(44*8, 9*20-10,"ตั้งค่า Jog Speed",WHITE);
  sprintf(text,"ค่า Jog Speed เดิม : %.4f",jog_value);
  drawtext(30*8,10*20,"ค่า Jog Speed = ",WHITE);

  strcpy(text,"");getstring(text,46,9,10,NUMBER,LIGHTRED);
  if(strcmp(text,"")!=0)
    jog_value=atof(text);
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
}

void save_project(void)
{
  int i,c;

  framebox(20, 7, 80/*–20-*/, 14, DARKGRAY);
  drawtext(40*8, 9*20-10,"บันทึกข้อมูล Project ไฟล์",WHITE);

  drawtext(25*8,11*20,"ใส่ชื่อไฟล์ : ",WHITE);
  strcpy(text,"");getstring(text,36,10,8,ENGLISH,LIGHTRED);
  if(strcmp(text,"")!=0)
  {
    strcpy(prjfile,path_data_file);
    strcat(prjfile,text);
    strcat(prjfile,".prj");
  }
  if((proj=fopen(prjfile,"wt"))==NULL)
  {
    closegraph();
    printf("Error : Can't open project file. \n");
    exit(0);
  }else
  {
    fprintf(proj,"01 :Unit :%d\n",unit);
    fprintf(proj,"02 :ObjW :%.20f\n",objectwidth);
    fprintf(proj,"03 :ObjL :%.20f\n",objectlength);
    fprintf(proj,"04 :ObjH :%.20f\n",objectheight);
    for(i=0,c=5;i<TOOLMAX;i++)
    {
      fprintf(proj,"%02d :To-S :%.20f\n",c,toolsize[i]); c++;
      fprintf(proj,"%02d :To-D :%.20f\n",c,tooldeep[i]); c++;
      fprintf(proj,"%02d :To-N :%d\n"   ,c,toolno[i]);   c++;
    }
    fprintf(proj,"%2d :F-Rt :%.20f\n",c,feedrate_value); c++;
    fprintf(proj,"%2d :S-Sp :%.20f\n",c,spindle_value);  c++;
    fprintf(proj,"%2d :J-Vl :%.20f\n",c,jog_value);      c++;
    fprintf(proj,"%2d :S-ft :%d\n",c,fastspeed);         c++;
    fprintf(proj,"%2d :S-sl :%d\n",c,slowspeed);         c++;
    fprintf(proj,"%2d :S-cv :%d\n",c,curvespeed);        c++;
    fprintf(proj,"%2d :fDxf :%s\n",c,dxffile);           c++;
  }
  fclose(proj);
}

int show_project_onscreen(void)
{
  FILE *fp;
  int i,c;

  if((fp=fopen(prjfile,"rt"))==NULL)
  {
    return(NO);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  }
  {
    fgets(text,255,fp);    i=10;     unit=(int)(readnum(text,&i));
    fgets(text,255,fp);    i=10;    objectwidth =readnum(text,&i);
    fgets(text,255,fp);    i=10;    objectlength=readnum(text,&i);
    fgets(text,255,fp);    i=10;    objectheight=readnum(text,&i);
    for(c=0;c<TOOLMAX;c++)
    {
      fgets(text,255,fp);  i=10;    toolsize[c]=readnum(text,&i);
      fgets(text,255,fp);  i=10;    tooldeep[c]=readnum(text,&i);
      fgets(text,255,fp);  i=10;    toolno[c]  =(int)readnum(text,&i);
    }
    fgets(text,255,fp);    i=10;    feedrate_value = readnum(text,&i);
    fgets(text,255,fp);    i=10;    spindle_value  = readnum(text,&i);
    fgets(text,255,fp);    i=10;    jog_value      = readnum(text,&i);
    fgets(text,255,fp);    i=10;    fastspeed      = (int)readnum(text,&i);
    fgets(text,255,fp);    i=10;    slowspeed      = (int)readnum(text,&i);
    fgets(text,255,fp);    i=10;    curvespeed     = (int)readnum(text,&i);
    fgets(text,255,fp);    i=10;    c=0;while(text[i]!='\n')
                                                    {dxffile[c]=text[i];
                                                     c++;i++;
                                                    }dxffile[c]=0;
    if(strcmp(dxffile,"")!=0)
    {
      read_file_dxf();
    }


    InitObject(objectwidth,objectlength,objectheight);
    fclose(fp);
    return(YES);
  }
}

void load_project(void)
{
  strcpy(prjfile,path_data_file);
  strcat(prjfile,"*.prj");
  ch=selectfile(prjfile);
  if(ch==YES)
  {
    if(show_project_onscreen()==YES)
    {
      //sprintf(text,"อ่าน Proj : %s",prjfile);

      drawtextbackground(27*8, 14*20,"อ่านค่าใน project file แล้ว กด key ใดๆ เพื่อทำงานต่อ",WHITE,GREEN,YELLOW);
    }else{
      drawtextbackground(27*8, 14*20,"ไม่สามารถเปิดไฟล์ project ได้",WHITE,GREEN,YELLOW);
    }

    getch();
    /*-------------------------------------*/
    /*      return view                    */
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    /*-------------------------------------------*/

    maxx=getmaxx();
    maxy=getmaxy();
    setviewport(0,0,maxx,maxy,1);
    //clearviewport();
  } else {
    drawtextbackground(27*8, 14*20,"ไม่มี Project ไฟล์ใน directory. สร้างไฟล์ dxf ก่อน",WHITE,GREEN,YELLOW);
    getch();
  }
  cleardevice();
}

void load_dxf(void)
{
  strcpy(dxffile,path_data_file);
  strcat(dxffile,"*.dxf");
  ch=selectfile(dxffile);

  if(ch==YES)
  {
    if(show_view_dxf_on_screen()==YES)
    {
      for(i=0;i<TOOLMAX;i++)
      {
            objectdeep[i]=0.0;
            toolno[i]=1;
      }
      getvalue_deep();
    } else
    {
      drawtextbackground(37*8, 15*20-10,"Can't open file name.",WHITE,GREEN,YELLOW);
      getch();
    }
    /*-------------------------------------------*/
    /*          return view                      */
    /*-------------------------------------------*/
    maxx=getmaxx();
    maxy=getmaxy();
    setviewport(0,0,maxx,maxy,1);
    //clearviewport();
  } else {
    drawtextbackground(37*8, 15*20-10,"ไม่มีไฟล์ DXF อยู่ใน directory. สร้างไฟล์ dxf ก่อน",WHITE,GREEN,YELLOW);
    getch();
  }
  cleardevice();
}

/*-------------------------------------------*/
void getvalue_deep(void)
{
  int ch,key;
  int col;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
char text[255];
int i;
int nobject = currentlayer.num;
double deep;
double object_test,deep_test;

do{
 for(i=0;i<nobject;i++)
 {
  setcolor(i+1);
  setfillstyle(SOLID_FILL,i+1);
  if(unit==INCH)
  {
        deep_test=objectdeep[i]/scaleunit;
  } else
  {
        deep_test=objectdeep[i];
  }
  sprintf(text," %2d  %2d     %6.2f   %8.3f",i+1,toolno[i],toolsize[i],deep_test);

  dispblank_xy_text((25+10),30+i*22,36,BLACK);
  drawtext((25+10),30+i*22,text,WHITE); //i+1);

  bar(10+10,(i)*22+30,25+10,(i+1)*22+30);
  setcolor(WHITE);
  rectangle(10+10,(i)*22+30,25+10,(i+1)*22+30);
 }

 drawtext(20+10,10+7*22,"N-ถัดไป P-ย้อนหลัง D-ใส่ความลึก Esc- ยกเลิก",WHITE);
 drawtext(20+10,10+8*22,"เลือก (N,P,D,Esc) : ",WHITE);
 do{
  ch=bioskey(0);
  key=ch&0x00ff;
  if (('a' <= key) && (key <= 'z'))
        key = key - ('a' - 'A');
 } while( (key!='N')&&(key!='P')&&(key!='D')&&(ch!=ESCKEY) );
 switch(key)
 {
  case 'N':

        break;
  case 'P':

        break;
  case 'D':
        col = BLACK;
        dispblank_xy_text(20+10,10+7*22,37,col);
        dispblank_xy_text(20+10,10+8*22,37,col);

        drawtext(30+10,8*22,"เลือกชั้น : ",WHITE);
        strcpy(text,"");getstring(text,14,8,2,NUMBER,LIGHTRED);

        i=atoi(text);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            strcpy(text,"ลึก :              ");
            if(unit==INCH)
              strcat(text,"นิ้ว");
            else
              strcat(text,"มิล.");
            drawtext(130+10,8*22,text,WHITE);
            strcpy(text,"");getstring(text,24,8,8,NUMBER,LIGHTRED);

            dispblank_xy_text(20,8*22,37,col);

            if( strcmp(text,"")!=0 )
            {
              if(i>0)
              {
                deep = atof(text);
                object_test = objectheight;
                if(unit==INCH)
                  deep_test   = deep * scaleunit;
                else
                  deep_test   = deep;

                if(deep_test>object_test)
                {
                  dispblank_xy_text(20+10,10+8*22,37,col);
                  drawtext       (20+10,8*22,"ความลึกเกิน กำหนดใหม่",WHITE);
                  getch();
                  dispblank_xy_text(20+10,8*22,37,BLACK);
                }
                else
                {
                  objectdeep[i-1]=deep_test;
                }
              }
            }
            break;
    }
  }while(ch!=ESCKEY);
}

void jog_mode(void)
{
  int ch,chf;
  double step=1.0;
  int hide=OFF;
  double mx,my,mz;

  SIMGRAPH=ON;

  do{
    if(hide==OFF)
    {
      framebox(20+10, 7, 80-20+10, 15, DARKGRAY);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    drawtext((34+10)*8, 9*20-10,"เคลื่อนที่ X Y Z",WHITE);

    drawtext((22+10)*8,10*20-10,"    Left  X+  Up   Y+  PgUp Z+ ",WHITE);
    drawtext((22+10)*8,11*20-10,"    Right X-  Down Y-  PgDw Z- ",WHITE);
    drawtext((22+10)*8,12*20-10,"    X,Y,Z เลือกแกนเคลื่อนที่ ",WHITE);
    drawtext((22+10)*8,13*20-10,"    + เพิ่มขั้นเลื่อน  , - ลดขั้นเลื่อน",WHITE);
    drawtext((22+10)*8,15*20-10,"    Esc - กลับเมนูหลัก",WHITE);
    dispblank_xy_text((46+10)*8,14*20-10,10,GREEN);
    sprintf(text,"    ขั้นในการเลื่อนแนวแกน : %.4f",step);
    drawtext((22+10)*8,14*20-10,text,WHITE);
  } else
  drawtextbackground((5+10)*8, 20*20-10,"ESC - ออกจาก Jog Mode",WHITE,GREEN,YELLOW);

//  abs_pos_x=toolx;   abs_pos_y=tooly;   abs_pos_z=toolz;
  out_position();
  ch=getch();
  TIME=6;
  switch(ch)
  {
    case 0:
          chf=getch();
          switch(chf)
          {
            case KEY_LEFT :
                    MoveLineTo(toolx+step , tooly      ,toolz      );
              break;
            case KEY_RIGHT:
                    MoveLineTo(toolx-step , tooly      ,toolz      );
              break;
            case KEY_UP   :
                    MoveLineTo(toolx      , tooly +step ,toolz      );
              break;
            case KEY_DOWN :
                    MoveLineTo(toolx      , tooly -step ,toolz      );
              break;
            case KEY_PGUP :
                    MoveLineTo(toolx      , tooly      ,toolz +step );
              break;
            case KEY_PGDW :
                    MoveLineTo(toolx      , tooly      ,toolz -step );
              break;
            case KEY_HOME :
                    if(hide==ON)
                    {
                      hide=OFF;
                      cleardevice();
                      frame();
                      Menubar(6);

                    }
                    else
                    {
                      hide=ON;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                    cleardevice();
                    frame();
                    Menubar(6);

                  }
          break;
        }
        break;
case 'x':case 'X':
          dispblank_xy_text((22+10)*8,15*20-12,30,GREEN);
          drawtext((22+10)*8,15*20-12,"เคลื่อนที่แกน x ไปตำแหน่ง : ",WHITE);
          strcpy(text,"");getstring(text,(46+10),13,10,NUMBER,LIGHTRED);
          if(strcmp(text,"")!=0)
            mx=atof(text);
          If(unit==INCH)
          {
            mx=mx*254;
          }
          MoveLineTo(mx ,tooly ,toolz );
break;

case 'y':case 'Y':
          dispblank_xy_text((22+10)*8,15*20-12,30,GREEN);
          drawtext((22+10)*8,15*20-12,"เคลื่อนที่แกน y ไปตำแหน่ง : ",WHITE);
          strcpy(text,"");getstring(text,(46+10),13,10,NUMBER,LIGHTRED);
          if(strcmp(text,"")!=0)
            my=atof(text);
          if(unit==INCH)
          {
            my=my*254;
          }
          MoveLineTo(toolx ,my ,toolz );

 break;

 case 'z':case 'Z':
          dispblank_xy_text((22+10)*8,15*20-12,30,GREEN);
          drawtext((22+10)*8,15*20-12,"เคลื่อนที่แกน z ไปตำแหน่ง : ",WHITE);
          strcpy(text,"");getstring(text,(46+10),13,10,NUMBER,LIGHTRED);
          if(strcmp(text,"")!=0)
            mz=atof(text);
          if(unit==INCH)
          {
            mz=mz*254;
          }
          MoveLineTo(toolx ,tooly ,mz );
 break;

 case '+':
        step++;     break;

 case '-':
        step--;if(step>1)step=1;     break;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

```
    }
  } while(ch!=ESC);
}

void draw_head(void)
{
  int col=getcolor();

  setcolor(WHITE);
        .
  line3df( toolx-15,tooly   ,toolz   ,
                toolx+15,tooly   ,toolz   ,WHITE);
  line3df( toolx   ,tooly+15,toolz   ,
                toolx   ,tooly-15,toolz   ,WHITE);
  line3df( toolx   ,tooly   ,toolz+15,
                toolx   ,tooly   ,toolz-15,WHITE);

  setcolor(col);
}

void get_value_feedrate(void)
{
  framebox(20, 7, 80, 13, DARKGRAY);
  drawtext(40*8, 9*20-10,"ค่า Feed Rate ที่ใช้ในโปรแกรม",WHITE);

  sprintf(text,"ความเร็วปกติ (%5d) = ",fastspeed);
  drawtext(25*8,10*20,text,WHITE);
  sprintf(text,"ความเร็วกัดแนวตรง (%5d) = ",slowspeed);
  drawtext(25*8,11*20,text,WHITE);
  sprintf(text,"ความเร็วกัดแนวโค้ง (%5d) = ",curvespeed);
  drawtext(25*8,12*20,text,WHITE);
  strcpy(text,"");getstring(text,46,9,10,NUMBER,LIGHTRED);
  if(strcmp(text,"")!=0)
    fastspeed=atoi(text);
  strcpy(text,"");getstring(text,51,10,10,NUMBER,LIGHTRED);
  if(strcmp(text,"")!=0)
    slowspeed=atoi(text);
  strcpy(text,"");getstring(text,51,11,10,NUMBER,LIGHTRED);
  if(strcmp(text,"")!=0)
    curvespeed=atoi(text);

  }

  void save_machine_code(void)
  {
    int c;
    FILE *fp1,*fp2;
    char savename[10];


    framebox(20, 7, 80/*--20-*/, 14, DARKGRAY);
    drawtext(38*8, 9*20-10,"บันทึกโปรแกรมควบคุมเครื่องจักร",WHITE);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
drawtext(25*8,11*20,"ใส่ชื่อไฟล์ :              .cod",WHITE);
strcpy(tmpname,"");getstring(tmpname,36,10,8,ENGLISH,LIGHTRED);
if(strcmp(tmpname,"")==0)
{
  strcpy(tmpname,mchname);
}

if(strcmp(tmpname,mchname)==0)
  return;

strcpy(savename,tmpname);
strcpy(text,path_data_file);
strcat(text,tmpname);
strcat(text,".cod");

strcpy(tmpname,path_data_file);
strcat(tmpname,mchname);
strcat(tmpname,".cod");

if((fp1=fopen(tmpname,"rb"))==NULL)
{
  printf("\n\n///Error : 0110");
  exit(0);
}

if((fp2=fopen(text,"wb"))==NULL)
{
  printf("\n\n///Error : 0111");
  exit(0);
}

do{
  ch=fgetc(fp1);
  fputc(ch,fp2);
}while(!feof(fp1));

fclose(fp1);
fclose(fp2);

strcpy(mchname,savename);
}

void load_machine(void)
{
  int i,j;

  strcpy(tmpname,path_data_file);
  strcat(tmpname,"*.cod");
  ch=selectfile(tmpname);

  if(ch==YES)
  {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
i=strlen(tmpname);
while( (tmpname[i]!='\\')&&(i>-1) )
  i--;
i++;j=0;
while(tmpname[i]!='.')
{
  mchname[j]=tmpname[i];
  j++;i++;
}
mchname[j]=0;
sprintf(text,"อ่านโปรแกรมควบคุมเครื่องจักร : %s.cod",mchname);
drawtextbackground(27*8, 14*20,text,WHITE,GREEN,YELLOW);
getch();
/*-------------------------------------*/
/*          return view                */
/*-------------------------------------*/

 maxx=getmaxx();
 maxy=getmaxy();
 setviewport(0,0,maxx,maxy,1);

} else {
 drawtextbackground(27*8, 14*20,"ไม่ได้เลือกโปรแกรมควบคุมเครื่องจักร กดปุ่มใดๆ เพื่อทพงานต่อ",WHITE,GREEN,YELLOW);
 getch();
}
 cleardevice();
}

void main(void)
{

 int sel;

 init_system();

 strcpy(mchname,"mch");
 set_pos_x=set_pos_y=set_pos_z=0.0;

 InitBuffer();

 nosound();


 Initialize_graph();  /* Initialize Graphics mode */

 initthaifont();
 setgraphmode(3);
 setColor256();
 maxx=getmaxx();
 maxy=getmaxy();

 objectwidth=300.0;
 objectlength=300.0;
```

เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
objectheight=200.0;
init_menu();

homex=tablex=3000.0;
homey=tabley=3000.0;
homez=tablez=2000.0;
xp=0.0  ;  yp=0.0  ;  zp=50.0;
degx=30.0;  degy=-214.0;  degz=-44.0;
init3d(degx,degy,degz);

init3d_all_view();
InitObject(objectwidth,objectlength,objectheight);

set_pos_x=objectx=tablex/2-objectwidth/2;
set_pos_y=objecty=tabley/2-objectlength/2;
set_pos_z=objectz=tablez/2-objectheight;

toolx=homex;
tooly=homey;
toolz=homez;

view1=view2=view3=view4=ON;
*dn_sel=1;

do{
  cleardevice();
  frame();

  do
  {
    sel=Selection(pop_menu[mainmenu],dn_sel,&row[mainmenu]);
    if (sel==0)
            mainmenu=*dn_sel;

    cleardevice();
    frame();
  } while(sel==0);

  Menubar(sel);
  frame();

  switch(sel)
  {
          case 1:     switch(*dn_sel)
                      {
                        case 1: /*---------- open file ----------*/
                                save_project();          break;

                        case 2: /*- read project file -*/
                                load_project();          break;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        case 3: /*- read dxf file -*/
                load_dxf();                     break;

        case 4:
                closegraph();
                quit=ON;
                exit(0);        break;
      } break;

case 2: switch(*dn_sel)  /*- new object and tools -*/
      {
        case 1: /*-  input size of object -*/
                framebox(30, 7, 80-10, 14, DARKGRAY);
                drawtext((34+10)*8, 9*20-10,"ใส่ขนาดชิ้นงาน",WHITE);


                if(unit==INCH)
                {
                 dispobjx=objectwidth/scaleunit;
                 dispobjy=objectlength/scaleunit;
                 dispobjz=objectheight/scaleunit;
                }
                else
                {
                 dispobjx=objectwidth;
                 dispobjy=objectlength;
                 dispobjz=objectheight;
                }

                sprintf(text,"ความกว้าง (%8.4f) ",dispobjx);
                if(unit==INCH)
                  strcat(text,"นิ้ว : ");
                else
                  strcat(text,"มม.: ");
                drawtext((22+10)*8,10*20,text,WHITE);

                sprintf(text,"ความยาว  (%8.4f) ",dispobjy);
                if(unit==INCH)
                  strcat(text,"นิ้ว : ");
                else
                  strcat(text,"มม.: ");
                drawtext((22+10)*8,11*20,text,WHITE);

                sprintf(text,"ความหนา  (%8.4f) ",dispobjz);
                if(unit==INCH)
                  strcat(text,"นิ้ว : ");
                else
                  strcat(text,"มม.: ");
                drawtext((22+10)*8,12*20,text,WHITE);


                strcpy(text,"");getstring(text,47+10, 9,10,NUMBER,LIGHTRED);
                if( strcmp(text,"")!=0 )
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                {
                 dispobjx=atof(text);
                 if(unit==INCH)
                 {
                   objectwidth=dispobjx*scaleunit;
                 } else
                 {
                   objectwidth=dispobjx;
                 }
                }
                strcpy(text,"");getstring(text,47+10,10,10,NUMBER,LIGHTRED);
                if( strcmp(text,"")!=0 )
                {
                 dispobjy=atof(text);
                 if(unit==INCH)
                 {
                   objectlength=dispobjy*scaleunit;
                 } else
                 {
                   objectlength=dispobjy;
                 }
                }

                strcpy(text,"");getstring(text,47+10,11,10,NUMBER,LIGHTRED);
                if( strcmp(text,"")!=0 )
                {
                 dispobjz=atof(text);
                 if(unit==INCH)
                 {
                   objectheight=dispobjz*scaleunit;
                 } else
                 {
                   objectheight=dispobjz;
                 }
                }

                break;

            case 2:
                getvalue_tools();
                  break;

          }
          InitObject(objectwidth,objectlength,objectheight);
          break;
case 3:   switch(*dn_sel)  /*-   dxf deep   -*/
          {
            case 1:
                if( show_view_dxf_on_screen()==YES)
                {
                  getvalue_deep();
                  getch();
                }else
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                {
                 framebox(30, 7, 80-10, 14, DARKGRAY);
                 drawtext((32+10)*8, 9*20-10,"กำหนดชั้นความลึก",WHITE);
                 drawtext((25+10)*8, 10*20-10,"ยังไม่มีข้อมูลภาพ dxf",WHITE);
                 drawtext((25+10)*8, 11*20-10,"ใส่ข้อมูล dxf ก่อน",WHITE);
                 drawtext((25+10)*8, 14*20-10,"Press any key to continue.",WHITE);
                 getch();
                }
                /*-------------------------------------------*/
                /*            return view                    */
                /*-------------------------------------------*/
                maxx=getmaxx();
                maxy=getmaxy();
                setviewport(0,0,maxx,maxy,1);
                clearviewport();

                  break;
        case 2:
                framebox(20+10, 7, 80-20+10, 14, DARKGRAY);
                drawtext((32+10)*8, 9*20-10,"สร้างโปรแกรม (อัตโนมัติ)",WHITE);

                if(currentlayer.num==0)
                {

                 drawtext((25+10)*8, 10*20-10,"ยังไม่มีข้อมูลภาพ dxf",WHITE);
                 drawtext((25+10)*8, 11*20-10,"ใส่ข้อมูล dxf ก่อน",WHITE);
                 drawtext((25+10)*8, 14*20-10,"Press any key to continue.",WHITE);
                 getch();
                 break;
                }
                draw_dxf_fill(SHOWCUT);
                compile_code();

                drawtext((25+10)*8, 12*20-10,"สร้างโปรแกรมควบคุมเครื่องจักรเสร็จแล้ว",WHITE);
                drawtext((28+10)*8, 14*20-10,"Press any key to continue.",WHITE);
                getch();
                  break;
          case 3:get_value_feedrate();   break;
          case 4:save_machine_code();    break;
          case 5:load_machine();         break;

        } break;
case 4: switch(*dn_sel)
        {
          case 1: TIME=2;
                  SIMGRAPH=ON;
                  FlagAction=DO_SIM;
                  simulate();
                  ViewSolidObject();
                 // draw_dxf_3d();
                  drawtextbackground(37*8, 14*20,"Press any key to continue.",WHITE,GREEN,YELLOW);
                  getch();
                  break;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
case 2: TIME=0;
                SIMGRAPH=ON;
                FlagAction=DO_SIM;
                simulate();
                ViewSolidObject();
               //  draw_dxf_3d();
                drawtextbackground(37*8, 14*20,"Press any key to continue.",WHITE,GREEN,YELLOW);
                getch();
                break;
 case 3: TIME=0;
                view1=view2=view3=view4=OFF;
                view5=ON;
                SIMGRAPH=OFF;
               // simulate();
                block3dhide(0.0,0.0,0.0,objectwidth,objectlength,-objectheight,WHITE);
                ViewSolidObject();
                block3dhide(0.0,0.0,0.0,objectwidth,objectlength,-objectheight,WHITE);
               //  draw_dxf_3d();
                view1=view2=view3=view4=ON;
                view5=OFF;
                drawtextbackground(37*8, 25*20,"Press any key to continue.",WHITE,GREEN,YELLOW);
                getch();
                break;
 } break;

case 5:  switch(*dn_sel)
 {

  case 1:set_feed_rate();     break;
  case 2:set_spindle_rate();  break;
  case 3:set_jog_speed();     break;
  case 4:toggle();                  break;

 } break;
case 6:  switch(*dn_sel)
 {

   case 1:

          jog_mode();

            break;
   case 2:
          framebox(20+10, 7, 80-20+10, 14, DARKGRAY);
          drawtext((34+10)*8, 9*20-10,"HOME X Y Z",WHITE);
          strcpy(text,"เคลื่อนที่แกน x,y,z ไปยังจุดเริ่มต้น");
          drawtext((22+10)*8,10*20,text,WHITE);


          MoveLineTo(homex,homey,homez);

          strcpy(text,"อยู่ที่จุดเริ่มต้นแล้ว");
          drawtext((22+10)*8,12*20,text,WHITE);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                    strcpy(text,"Press any key to continue.");
                    drawtext((22+10)*8,13*20,text,WHITE);

                    getch();
                      break;
                case 3:

                    framebox(20+10, 7, 80-20+10, 14, DARKGRAY);
                    drawtext((30+10)*8,11*20-10,"ตั้งค่าศูนย์ X Y Z ใหม่",WHITE);

                    getch();
                      break;
              } break;
        case 7:  switch(*dn_sel)
              {

                case 1:
                    FlagAction=DO_CYCLE;
                    simulate();

                    getch();
                      break;
                case 2: /*- cycle stop -*/ break;
                case 3:
                    setfillstyle(SOLID_FILL,RED);
                    bar(100,100,200,200);
                    setcolor(WHITE);
                    outtextxy(120,120,"7 , 3 ");
                      break;
                case 4:
                    setfillstyle(SOLID_FILL,RED);
                    bar(100,100,200,200);
                    setcolor(WHITE);
                    outtextxy(120,120,"7 , 4 ");
                      break;
              }
          break;
    }
    *dn_sel = sel;

  }while(quit==OFF);

}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติผู้เขียน

นายรวิ อุตตมธนินทร์ เกิดเมื่อวันที่ 23 สิงหาคม 2513 ที่จังหวัดระนอง เริ่มเข้าโรงเรียนชั้นอนุบาลที่โรงเรียนน่านคริสเตียนศึกษา จังหวัดน่าน ได้ย้ายไปเรียนระดับประถมที่โรงเรียนเชียงรายวิทยาคม จังหวัดเชียงราย ต่อมาได้ย้ายเข้าเรียนที่กรุงเทพมหานคร เข้ามาเรียนที่โรงเรียนเรวดีในระดับชั้น ป.4-ป.6 และในชั้นมัธยมต้นเรียนที่โรงเรียนสามเสนวิทยาลัย และในชั้น ม.ปลายได้เรียนที่โรงเรียนเตรียมอุดมศึกษา และได้สอบเทียบจบชั้น ม. 6 ที่โรงเรียนสุรศักดิ์มนตรี และได้สำเร็จการศึกษาวิศวกรรมศาสตร์บัณฑิต(คอมพิวเตอร์) จากสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ปีการศึกษา 2533 เข้าทำงานในตำแหน่งอาจารย์ประจำภาควิชาวิศวกรรมคอมพิวเตอร์ มหาวิทยาลัยมหานคร หนองจอก ตั้งแต่ปี 2534 ปัจจุบันทำงานอิสระ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้