การพัฒนาโปรแกรมคอมพิวเตอร์ช่วยในการผลิตร่วมกับออโตแคดโดยใช้ ARX

นายมนูศักดิ์ จานทอง

หอสมุดกลาง สถาบันวิทยบริการ จุฬาลงกรณ์มหาวิทยาลัย

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมเครื่องกล ภาควิชาวิศวกรรมเครื่องกล
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2542

ISBN 974-334-739-9

ลิขสิทธิ์ของ จุฬาลงกรณ์มหาวิทยาลัย

I19377563

DEVELOPMENT A COMPUTER-AIDED MANUFACTURING SOFTWARE INTERFACE WITH AUTOCAD BY USING ARX

Mr. Manusak Janthong

A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Engineering in Mechanical Engineering
Department of Mechanical Engineering
Faculty of Engineering
Chulalongkorn University
Academic Year 1999

ISBN 974-334-739-9

หัวข้อวิทยานิพนธ์	การพัฒนาโปรแกรมคอมพิวเตอร์ช่วยในการผลิตร่วมกับออโตแคด โดยใช้ ARX
โดย	นายมนูศักดิ์ จานทอง
ภาควิชา	วิศวกรรมเครื่องกล
อาจารย์ที่ปรึกษา	รองศาสตราจารย์ ดร. วิบูลย์ แสงวีระพันธุ์ศิริ

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้นับวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญามหาบัณฑิต

..... คณบดีคณะวิศวกรรมศาสตร์
(ศาสตราจารย์ ดร.สมศักดิ์ ปัญญาแก้ว)

คณะกรรมการสอบวิทยานิพนธ์

..... ประธานกรรมการ
(รองศาสตราจารย์ ดร. วิทยา ยงเจริญ)

..... อาจารย์ที่ปรึกษา
(รองศาสตราจารย์ ดร. วิบูลย์ แสงวีระพันธุ์ศิริ)

..... กรรมการ
(รองศาสตราจารย์ ดร. สมศักดิ์ ไชยะภินันท์)

..... กรรมการ
(ผู้ช่วยศาสตราจารย์ ดร. ชัยโรจน์ คุณพนิชกิจ)

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

มนูศักดิ์ จานทอง : การพัฒนาโปรแกรมคอมพิวเตอร์ช่วยในการผลิตร่วมกับออโตแคด โดยใช้ ARX. (DEVELOPMENT OF A COMPUTER-AIDED MANUFACTURING SOFTWARE INTERFACE WITH AUTOCAD BY USING ARX)
อ. ที่ปรึกษา : รศ.ดร. วิบูลย์ แสงวีระพันธุ์ศิริ, 90 หน้า. ISBN 974-334-739-9.

ในงานวิจัยนี้เป็นการพัฒนาโปรแกรมคอมพิวเตอร์ช่วยในการผลิต โดยโปรแกรมที่พัฒนาจะคำนวณหาเส้นทางเดินของหัวกัด (Tool path) แล้วแปลงเส้นทางเดินของหัวกัดให้เป็นรหัสจี (G-Code) ที่ใช้สำหรับเครื่องกัดซีเอ็นซี (CNC Milling Machine) และทำงานร่วมกับโปรแกรมออโตแคดรีลีส 14 โดยใช้ชุดตัวต่อประสาน (Interface) ARX ช่วยในการเข้าถึงข้อมูลภายในแบบชิ้นงานที่วาดจากโปรแกรมออโตแคดและใช้ในการคำนวณหาเส้นทางเดินของหัวกัด โปรแกรมที่พัฒนาแบ่งออกเป็น 5 โปรแกรมย่อยด้วยกัน ดังนี้ 1.โปรแกรมสำหรับงานเจาะ 2. โปรแกรมสำหรับงานกัดหยาบแบบ 2 แกนครึ่ง 3. โปรแกรมสำหรับงานกัดละเอียดแบบ 2 แกนครึ่ง 4. โปรแกรมสำหรับงานกัดหยาบแบบ 3 แกน 5. โปรแกรมสำหรับงานกัดละเอียดแบบ 3 แกน

ในการทดสอบโปรแกรมที่พัฒนาผู้วิจัยได้นำรหัสจีที่ได้จากโปรแกรมที่พัฒนาไปกัดชิ้นงานจริงด้วยเครื่องกัดซีเอ็นซีของ FANUC Model MA15 และวัสดุที่ใช้ในการทดสอบได้แก่ อลูมิเนียม ไม้ และไข (Wax) เป็นต้น โดยผลที่ได้จากการทดสอบก็คือ มีค่าความผิดพลาดหลังจากการวัดขนาดอยู่ในช่วง 10 ถึง 20 ไมครอนสำหรับโปรแกรมสำหรับงานเจาะ ความผิดพลาดในช่วง 20 ถึง 70 ไมครอนสำหรับโปรแกรมสำหรับงานกัดละเอียดแบบ 2 แกนครึ่ง และความผิดพลาดในช่วง 100 ถึง 115 ไมครอน สำหรับโปรแกรมกัดละเอียดแบบ 3 แกน

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ภาควิชา วิศวกรรมเครื่องกล ลายมือชื่อนิสิต .....
สาขาวิชา วิศวกรรมเครื่องกล ลายมือชื่ออาจารย์ที่ปรึกษา .....
ปีการศึกษา 2542 ลายมือชื่ออาจารย์ที่ปรึกษาร่วม .....

3971346521: MAJOR MECHANICAL ENGINEERING
KEY WORD: CAM / ARX / OBJECTARX
MANUSAK JANTHONG: DEVELOPMENT OF A COMPUTER-AIDED MANUFACTURING SOFTWARE INTERFACE WITH AUTOCAD BY USING ARX.
THESIS ADVISOR: ASSO. PROF. VIBOON SANGVERAPHUNSIRI, Ph.D., 90 pp. ISBN 974-334-739-9.

This research is the development of a computer-aided manufacturing software for AutoCAD. It can be used in 2-D, 2½-D and 3-D milling processes including cutting simulation. The output files are G-Code files which can be used for CNC milling machines. The ARX interface is used for the program to interface with the database inside the AutoCAD release 14. The software package consists of 5 subprograms as the 2-D Drilling, 2½ -D Pocket Milling, 2½ - D Contour Milling, 3-D Pocket Milling and 3-D Contour Milling.

The Makino milling machine with FANUC MA15 controller is used for testing the program. Various materials such as Aluminum, Wood and Wax are used as raw materials. The experimental results shown that the error is in the range of 10-20 micron for the 2-D Drilling program, 20-70 micron for the 2½ -D Contour Milling program and 100-115 micron for the 3-D Contour Milling can be achieved.

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ภาควิชา วิศวกรรมเครื่องกล ลายมือชื่อนิสิต .....
สาขาวิชา วิศวกรรมเครื่องกล ลายมือชื่ออาจารย์ที่ปรึกษา .....
ปีการศึกษา 2542 ลายมือชื่ออาจารย์ที่ปรึกษาร่วม .....

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงไปได้ด้วยความช่วยเหลืออย่างดียิ่งของ รองศาสตราจารย์ ดร. วิบูลย์ แสงวีระพันธุ์ศิริ อาจารย์ที่ปรึกษาวิทยานิพนธ์ ซึ่งท่านได้ให้คำแนะนำและข้อคิดเห็นต่างๆ ในการวิจัยมาด้วยดีตลอด และขอขอบคุณบริษัท NSS จำกัด ที่มอบทุนการศึกษาให้โดยไม่มีเงื่อนไขใดๆ ทั้งสิ้น และเนื่องจากทุนการวิจัยครั้งนี้บางส่วนได้รับมาจากทุนอุดหนุนการวิจัยของบัณฑิตวิทยาลัย จึงขอขอบพระคุณบัณฑิตวิทยาลัยมา ณ ที่นี้ด้วย

ท้ายนี้ ผู้วิจัยใคร่ขอกราบขอพระคุณ บิดา-มารดา ซึ่งสนับสนุนในด้านการเงินและให้กำลังใจแก่ผู้วิจัยเสมอมาจนสำเร็จการศึกษา

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

สารบัญ

บทที่		หน้า
บทคัดย่อภาษาไทย		ง
บทคัดย่อภาษาอังกฤษ		จ
กิตติกรรมประกาศ		ฉ
สารบัญ		ช
สารบัญตาราง		ซ
สารบัญภาพ		ฌ
บทที่		
1.	บทนำ	1
	1.1 ความเป็นมาและความสำคัญของปัญหา	1
	1.2 วัตถุประสงค์ของการวิจัย	1
	1.3 ขอบเขตของการวิจัย	2
	1.4 ขั้นตอนการดำเนินวิจัย	2
	1.5 ประโยชน์ที่คาดว่าจะได้รับ	2
2.	ทฤษฎีที่เกี่ยวข้อง	3
	2.1 เทคโนโลยีเอ็นซีและซีเอ็นซี	3
	2.2 พื้นฐานของคอมพิวเตอร์ช่วยในการออกแบบและช่วยในการผลิต	16
	2.3 โปรแกรมออโดแคด (AutoCAD)	19
	2.4 ตัวต่อประสาน ARX (ARX Interface)	19
	2.5 การวาดรูปผิวด้วยโปรแกรมออโตแคด	19
3.	โครงสร้างและขั้นตอนการใช้งานโปรแกรม	23
	3.1 โครงสร้างและการออกแบบโปรแกรม	24
	3.2 การติดตั้งและการโหลด (Load) โปรแกรม ARX	34
	3.3 การใช้งานโปรแกรมที่พัฒนา	35
4.	การทดสอบโปรแกรมและผลการทดสอบ	53
5.	สรุปผลการวิจัย และข้อเสนอแนะ	68
รายการอ้างอิง		72
ภาคผนวก		73
	ภาคผนวก ก.	74
	ภาคผนวก ข.	80
ประวัติผู้เขียน		90

สารบัญตาราง

ตาราง		หน้า
2.1	แสดงค่าพิกัด XY ด้วยระบบตำแหน่งแบบสัมบูรณ์	13
2.2	แสดงค่าพิกัด XY ด้วยระบบตำแหน่งแบบต่อเนื่อง	14
2.3	แสดงความหมายของกลุ่มรหัส	15
ก.1	แสดงรายละเอียดข้อกำหนดเครื่องมือวัดพิกัดของ Brown & Sharpe รุ่น MicroVal 343	76

สารบัญภาพ

ภาพประกอบ		หน้า
2.1	รูปส่วนประกอบพื้นฐานหลักของเครื่องเอ็นซีและซีเอ็นซี	4
2.2	แสดงลักษณะการเคลื่อนที่แบบเดินในแนวแกน	5
2.3	แสดงลักษณะการเคลื่อนที่แบบเดินเป็นเส้นตรงทำมุม 45°	5
2.4	แสดงลักษณะการเคลื่อนที่แบบเดินเชิงเส้น	6
2.5	แสดงลักษณะการเคลื่อนที่แบบระบบต่อเนื่อง	7
2.6	แสดงลักษณะการประมาณค่าแบบเชิงเส้น	8
2.7	การประมาณค่าแบบเชิงวงกลมในระนาบต่างๆ	9
2.8	แสดงลักษณะการประมาณค่าแบบเชิงเกลียว	9
2.9	แสดงลักษณะการประมาณค่าแบบเชิงพาราโบลา	10
2.10	เครื่องกัดซีเอ็นซีแบบตั้งในระบบพิกัดฉาก	11
2.11	ระบบพิกัดฉาก	11
2.12	ระบบพิกัดเชิงขั้วแบบ 2 แกน	12
2.13	ระบบพิกัดเชิงขั้วแบบ 3 แกนหรือระบบพิกัดทรงกระบอก	12
2.14	ภาพการเจาะรูจำนวน 6 รู จาก H1 ถึง H6 ตามลำดับ	13
2.15	ภาพแสดงรูปภาพที่วาดขึ้นจากแบบจำลองโครงลวด	17
2.16	ภาพแสดงรูปภาพที่วาดขึ้นจากแบบจำลองรูปผิว	17
2.17	ภาพแสดงรูปภาพที่วาดขึ้นจากแบบจำลองโซลิด	18
2.18	ภาพแสดงโพลีกอนเมชเปิดและปิด	20
2.19	ภาพแสดงรูปผิวที่สร้างจากคำสั่ง 3D	20
2.20	ภาพแสดงความแตกต่างของค่า M และ N ของคำสั่ง 3Dmesh	21
2.21	ภาพแสดงการสร้างรูปผิวจากคำสั่ง RuleSurf	21
2.22	ภาพแสดงการสร้างรูปผิวจากคำสั่ง TabSurf	21
2.23	ภาพแสดงการสร้างรูปผิวจากคำสั่ง RevSurf	22
2.24	ภาพแสดงการสร้างรูปผิวจากคำสั่ง EdgeSurf	22
3.1	ภาพแสดงแผนผังของโปรแกรมสำหรับงานเจาะ	25
3.2	ภาพแสดงแผนผังของโปรแกรมสำหรับงานกัดหยาบแบบ 2 แกนครึ่ง	27
3.3	ภาพแสดงแผนผังของโปรแกรมสำหรับงานกัดละเอียดแบบ 2 แกนครึ่ง	29
3.4	ภาพแสดงโพลีกอนเมชจากการวาดรูปผิวด้วยโปรแกรมออโตแคด	30
3.5	ภาพแสดงการจัดเรียงจุดของการกัดละเอียด	31
3.6	ภาพแสดงแผนผังของโปรแกรมสำหรับงานกัดหยาบแบบ 3 แกน	32
3.7	ภาพแสดงแผนผังของโปรแกรมสำหรับงานกัดละเอียดแบบ 3 แกน	33

สารบัญภาพ (ต่อ)

ภาพประกอบ		หน้า
3.8	ภาพแสดงการพิมพ์ชื่อไฟล์ ARX ด้วยโปรแกรมโน้ตแพด	34
3.9	ภาพแสดงไดอะล็อกบ็อกซ์ของการโหลดไฟล์ ARX ด้วยตนเอง	35
3.10	ภาพแสดงไดอะล็อกบ็อกซ์แรกก่อนเข้าใช้โปรแกรมสำหรับงานเจาะ	36
3.11	ภาพแสดงไดอะล็อกบ็อกซ์ให้เติมชื่อไฟล์รหัสจี	36
3.12	ภาพแสดงไดอะล็อกบ็อกซ์ให้ใส่ข้อมูลของโปรแกรมสำหรับงานเจาะ	37
3.13	ภาพแสดงไดอะล็อกบ็อกซ์เพื่อเติมข้อมูลเกี่ยวกับดอกสว่าน	38
3.14	ภาพแสดงไดอะล็อกบ็อกซ์แรกก่อนเข้าใช้โปรแกรมสำหรับงานกัดหยาบ แบบ 2 แกนครึ่ง	40
3.15	แสดงไดอะล็อกบ็อกซ์สำหรับเลือกรูปแบบการกัดของโปรแกรมสำหรับ งานกัดหยาบแบบ 2 แกนครึ่ง	41
3.16	ภาพแสดงไดอะล็อกบ็อกซ์ให้ใส่ข้อมูลของโปรแกรมสำหรับงาน กัดหยาบแบบ 2 แกนครึ่ง	41
3.17	ภาพแสดงไดอะล็อกบ็อกซ์เพื่อเติมข้อมูลเกี่ยวกับหัวกัด	42
3.18	ภาพแสดงไดอะล็อกบ็อกซ์แรกก่อนเข้าใช้โปรแกรมสำหรับ งานกัดละเอียดแบบ 2 แกนครึ่ง	43
3.19	ภาพแสดงไดอะล็อกบ็อกซ์ให้ใส่ข้อมูลของโปรแกรมสำหรับ งานกัดละเอียดแบบ 2 แกนครึ่ง	44
3.20	ภาพแสดงไดอะล็อกบ็อกซ์เพื่อเติมข้อมูลเกี่ยวกับหัวกัด	44
3.21	ภาพแสดงไดอะล็อกบ็อกซ์ แรกก่อนเข้าใช้โปรแกรมสำหรับ งานกัดหยาบแบบ 3 แกน	46
3.22	ภาพแสดงไดอะล็อกบ็อกซ์เพื่อใส่ข้อมูลของโปรแกรมสำหรับ งานกัดหยาบแบบ 3 แกน	46
3.23	ภาพแสดงไดอะล็อกบ็อกซ์เพื่อเติมข้อมูลเกี่ยวกับหัวกัด	47
3.24	ภาพแสดงไดอะล็อกบ็อกซ์เพื่อเติมข้อมูลเกี่ยวกับระนาบ	47
3.25	ภาพแสดงไดอะล็อกบ็อกซ์เพื่อใส่ข้อมูลของโปรแกรมสำหรับ งานกัดละเอียดของการกัดแบบ 3 แกน	48
3.26	ภาพแสดงไดอะล็อกบ็อกซ์ให้ใส่ข้อมูลของโปรแกรมสำหรับ งานกัดละเอียดแบบ 3 แกน	49
3.27	ภาพแสดงไดอะล็อกบ็อกซ์เพื่อให้ใส่ข้อมูลเกี่ยวกับหัวกัด	50
3.28	ภาพแสดงไดอะล็อกบ็อกซ์เพื่อเลือกรูปแบบการเข้าและออกชิ้นงาน	51

สารบัญภาพ (ต่อ)

ภาพประกอบ		หน้า
4.1	ภาพแสดงแบบชิ้นงานที่ 1	54
4.2	ภาพชิ้นงานจริงจากแบบชิ้นงานที่ 1 ที่ได้จากกัด	55
4.3	ภาพแสดงแบบชิ้นงานที่ 2	56
4.4	ภาพชิ้นงานจริงจากแบบชิ้นงานที่ 2 ที่ได้จากกัด	57
4.5	ภาพแสดงแบบชิ้นงานที่ 3	58
4.6	ภาพชิ้นงานจริงจากแบบชิ้นงานที่ 3 ที่ได้จากกัด	59
4.7	ภาพแสดงแบบชิ้นงานที่ 4	60
4.8	ภาพชิ้นงานจริงจากแบบชิ้นงานที่ 4 ที่ได้จากกัด	61
4.9	ภาพแสดงแบบชิ้นงานที่ 5	62
4.10	ภาพชิ้นงานจริงจากแบบชิ้นงานที่ 5 ที่ได้จากกัด	62
4.11	ภาพแสดงแบบชิ้นงานที่ 6	63
4.12	ภาพชิ้นงานจริงจากแบบชิ้นงานที่ 6 ที่ได้จากกัด	64
4.13	ภาพแสดงแบบชิ้นงานที่ 7	65
4.14	ภาพชิ้นงานจริงจากแบบชิ้นงานที่ 7 ที่ได้จากกัด	65
4.15	ภาพแสดงแบบชิ้นงานที่ 8	66
4.16	ภาพชิ้นงานจริงจากแบบชิ้นงานที่ 8 ที่ได้จากกัด	66
4.17	ภาพแบบชิ้นงานที่ใช้ทดสอบโปรแกรมสำหรับงานกัดหยาบแบบ 3 แกน	67
4.18	ภาพแสดงชิ้นงานที่ถูกกัดด้วยโปรแกรมสำหรับงานกัดหยาบแบบ 3 แกน	67
5.1	แสดงแบบชิ้นงานตัวอย่างที่ไม่สามารถออฟเซตได้สมบูรณ์	69
ก.1	ภาพแสดงเครื่องกัดซีเอ็นซีที่ใช้ในการกัดชิ้นงาน	74
ก.2	ภาพแสดงเครื่องมือวัดพิกัดของ Brown & Sharpe รุ่น MicroVal 343	75
ก.3	แสดงภาพไมโครมิเตอร์แบบปลายบอล	77
ก.4	แสดงภาพไมโครมิเตอร์แบบปลายแฟลตวัดได้ในช่วง 0-25 มม.	77
ก.5	แสดงภาพไมโครมิเตอร์แบบปลายแฟลตวัดได้ในช่วง 25-50 มม.	78
ก.6	ภาพแสดงหัวกัดที่ใช้	79
ข.1	ภาพการจำลองเส้นทางเดินของหัวกัดของโปรแกรมสำหรับงานเจาะ ขณะเริ่มต้น	81
ข.2	ภาพการจำลองเส้นทางเดินของหัวกัดของโปรแกรมสำหรับงานเจาะ ขณะกำลังกัดชิ้นงาน	81

สารบัญภาพ (ต่อ)

ภาพประกอบ		หน้า
ข.3	ภาพการจำลองเส้นทางเดินของหัวกัดของโปรแกรมสำหรับงานเจาะ เมื่อจบจำลองเส้นทางเดินของหัวกัด	82
ข.4	ภาพการจำลองเส้นทางเดินของหัวกัดของโปรแกรมสำหรับงานกัดหยาบ แบบ 2 แกนครึ่งขณะเริ่มต้น	82
ข.5	ภาพการจำลองเส้นทางเดินของหัวกัดของโปรแกรมสำหรับงานกัดหยาบ แบบ 2 แกนครึ่งขณะกำลังกัดชิ้นงาน	83
ข.6	ภาพการจำลองเส้นทางเดินของหัวกัดของโปรแกรมสำหรับงานกัดหยาบ แบบ 2 แกนครึ่งเมื่อจบจำลองเส้นทางเดินของหัวกัด	83
ข.7	ภาพการจำลองเส้นทางเดินของหัวกัดของโปรแกรมสำหรับงานกัดละเอียด แบบ 2 แกนครึ่งขณะเริ่มต้น	84
ข.8	ภาพการจำลองเส้นทางเดินของหัวกัดของโปรแกรมสำหรับงานกัดละเอียด แบบ 2 แกนครึ่งขณะกำลังกัดชิ้นงาน	84
ข.9	ภาพการจำลองเส้นทางเดินของหัวกัดของโปรแกรมสำหรับงานกัดละเอียด แบบ 2 แกนครึ่งเมื่อจบจำลองเส้นทางเดินของหัวกัด	85
ข.10	ภาพการจำลองเส้นทางเดินของหัวกัดของโปรแกรมสำหรับงานกัดหยาบ แบบ 3 แกนขณะเริ่มต้น	85
ข.11	ภาพการจำลองเส้นทางเดินของหัวกัดของโปรแกรมสำหรับงานกัดหยาบ แบบ 3 แกนขณะกำลังกัดชิ้นงาน	86
ข.12	ภาพการจำลองเส้นทางเดินของหัวกัดของโปรแกรมสำหรับงานกัดหยาบ แบบ 3 แกนเมื่อจบจำลองเส้นทางเดินของหัวกัด	86
ข.13	ภาพการจำลองเส้นทางเดินของหัวกัดของโปรแกรมสำหรับงานกัดละเอียด แบบ 3 แกน วิ่งแบบ One-Way ขณะเริ่มต้น	87
ข.14	ภาพการจำลองเส้นทางเดินของหัวกัดของโปรแกรมสำหรับงานกัดละเอียด แบบ 3 แกน วิ่งแบบ One-Way ขณะกำลังกัดชิ้นงาน	87
ข.15	ภาพการจำลองเส้นทางเดินของหัวกัดของโปรแกรมสำหรับงานกัดละเอียด แบบ 3 แกน วิ่งแบบ One-Way เมื่อจบจำลองเส้นทางเดินของหัวกัด	88
ข.16	ภาพการจำลองเส้นทางเดินของหัวกัดของโปรแกรมสำหรับงานกัดละเอียด แบบ 3 แกน วิ่งแบบ Zig-Zag ขณะเริ่มต้น	88
ข.17	ภาพการจำลองเส้นทางเดินของหัวกัดของโปรแกรมสำหรับงานกัดละเอียด แบบ 3 แกน วิ่งแบบ Zig-Zag ขณะกำลังกัดชิ้นงาน	89

สารบัญภาพ (ต่อ)

ภาพประกอบ หน้า

ข.18 ภาพการจำลองเส้นทางเดินของหัวกัดของโปรแกรมสำหรับงานกัดละเอียด แบบ 3 แกน วิ่งแบบ Zig-Zag เมื่อจบจำลองเส้นทางเดินของหัวกัด 89

บทที่ 1
บทนำ

ความเป็นมาและความสำคัญของปัญหา

ปัจจุบันองค์กรธุรกิจและวงการอุตสาหกรรมส่วนใหญ่พยายามที่จะเพิ่มขีดความสามารถในการแข่งขัน โดยการใช้เทคโนโลยีในการผลิตที่ทันสมัย เนื่องด้วยการแข่งขันทางการค้าที่ในทุกวันนี้ยิ่งทวีความรุนแรงยิ่งขึ้น ดังนั้นโรงงานอุตสาหกรรมหลายๆ แห่ง เริ่มที่จะตระหนักถึง และได้มีการนำระบบคอมพิวเตอร์มาช่วยเพื่อลดต้นทุนในการทำงาน ซึ่งเป็นเหตุให้มีการนำระบบซอฟต์แวร์ CAD/CAM มาช่วยออกแบบผลิตภัณฑ์และช่วยในการผลิต

โปรแกรมออโตแคด (AutoCAD) เป็นซอฟต์แวร์หนึ่งที่ช่วยในการออกแบบผลิตภัณฑ์ที่รู้จักกันอย่างกว้างขวางในวงการอุตสาหกรรมไทย โปรแกรมออโตแคดเป็นซอฟต์แวร์ CAD (Computer Aided Design) ซึ่งใช้ช่วยเฉพาะในการออกแบบผลิตภัณฑ์เท่านั้น จึงไม่สามารถนำมาช่วยในการผลิตผลิตภัณฑ์ได้ ดังนั้นจึงจำเป็นจะต้องใช้ซอฟต์แวร์ CAM (Computer Aided Manufacturing) มาช่วยในการผลิต ซึ่งจะต้องแปลงแบบชิ้นงานที่วาดจากโปรแกรมออโต้แคดให้เก็บอยู่ในนามสกุล DXF แล้วนำไปเปิดใช้ในซอฟต์แวร์ CAM นั้น หรือไม่ก็ต้องตรวจก่อนว่าซอฟต์แวร์ CAM ซอฟต์แวร์นั้นสามารถรับไฟล์แบบชิ้นงานของโปรแกรมออโต้แคดได้โดยตรงหรือไม่ เพื่อจะได้เปิดไฟล์แบบชิ้นงานทำงานต่อไป จะเห็นว่ามีขั้นตอนในการทำงานที่ยุ่งยากและจะต้องใช้ซอฟต์แวร์ CAM เพิ่มอีกซอฟต์แวร์หนึ่ง ด้วยเหตุนี้ผู้วิจัยจึงได้ทำการพัฒนาซอฟต์แวร์ CAM เพิ่มเข้าไปในโปรแกรมออโตแคด โดยซอฟต์แวร์ CAM ที่พัฒนาเพิ่มให้กับโปรแกรมออโตแคดจะเป็นโปรแกรมช่วยในงานกัด (Milling) เท่านั้น ซึ่งโปรแกรมที่พัฒนานี้จะทำงานบนโปรแกรมออโต้แคด เสมือนเป็นฟังก์ชันหนึ่งของโปรแกรมออโต้แคด เพื่อช่วยลดขั้นตอนการทำงานที่ได้กล่าวมาแล้ว โปรแกรมออโตแคดเป็นซอฟต์แวร์ที่มีความยืดหยุ่นในตัวเอง ผู้ใช้สามารถแก้ไขและเพิ่มเติมรูปแบบหน้าตา และคำสั่งในการทำงานใหม่ๆ ในโปรแกรมออโตแคดได้ โดยโปรแกรมออโตแคดตั้งแต่ Release13 เป็นต้นมามีตัวประสาน (Interface) ที่ใช้ในการจัดการระบบภายใน เพิ่มประสิทธิภาพในการทำงานของโปรแกรมออโตแคดอยู่หลายประเภทด้วยกัน ได้แก่ AutoLisp, Visual Lisp, ADS (AutoCAD Development System) และ ARX (AutoCAD Runtime Extension) เป็นต้น ซึ่งในวิทยานิพนธ์ฉบับนี้ได้ใช้ตัวต่อประสาน ARX มาช่วยในการพัฒนาซอฟต์แวร์ CAM ให้กับโปรแกรมออโตแคด เพื่อทำให้โปรแกรมออโตแคดจะได้เป็นซอฟต์แวร์ CAD/CAM

วัตถุประสงค์ของการวิจัย

เพื่อพัฒนาโปรแกรมที่ใช้สำหรับสร้างรหัสจี (G-Code) ที่ใช้สำหรับเครื่องกัดซีเอ็นซี (CNC Milling Machine)

ขอบเขตของการวิจัย

1. สามารถเขียนรหัสจีสำหรับงานกัดรูปผิว (Surface) จากแบบชิ้นงานที่วาดขึ้นจากโปรแกรมออโตแคดรีลีส 14 (AutoCAD Release 14)

2. รหัสจีที่ได้จะเป็นรหัสจีของงานกัด 3 แกน (3D Contouring)

3. รหัสจีที่ได้จะใช้สำหรับเครื่องกัดซีเอ็นซีของ FANUC Model MA15

วิธีดำเนินการวิจัย

1. ศึกษาโครงสร้างและกระบวนการเขียนโปรแกรมด้วยตัวต่อประสาน ARX

2. ศึกษาการวาดแบบชิ้นงาน 3 แกน และการเก็บข้อมูล (Database) ของโปรแกรมออโตแคดรีลีส 14 ที่เป็นรูปผิว

3. ศึกษาขั้นตอนการทำงานของเครื่องกัดซีเอ็นซี

4. ศึกษาความหมายและการเขียนรหัสจีสำหรับเครื่องกัดซีเอ็นซี

5. ทดสอบโปรแกรมที่เขียนขึ้น โดยนำผลที่ได้จากโปรแกรมไปทดลองกัดชิ้นงานตัวอย่างบนเครื่องกัดซีเอ็นซี

6. ศึกษาข้อบกพร่องของโปรแกรมที่เขียนขึ้น และแก้ไขให้สามารถใช้งานได้มีประสิทธิภาพมากยิ่งขึ้น

7. ประเมินและสรุปผลการวิจัย

ประโยชน์ที่คาดว่าจะได้รับ

1. ได้ซอฟต์แวร์ CAM ที่สามารถทำงานบนโปรแกรมออโตแคด เพื่อช่วยลดขั้นตอนในการทำงานได้

2. สามารถสร้างรหัสจีสำหรับงานผิวจากแบบชิ้นงานที่เขียนขึ้นจากโปรแกรมออโตแคดรีลีส 14

3. สามารถพัฒนาโปรแกรมที่ต้องการเพิ่มความสามารถการใช้งานบนโปรแกรมออโตแคดรีลีส 14 ได้

4. เข้าใจถึงลักษณะการจัดเก็บข้อมูลของแบบชิ้นงานที่วาดขึ้นจากโปรแกรมออโตแคด

5. สามารถเขียนโปรแกรม ARX รับข้อมูลของแบบชิ้นงานที่วาดขึ้นจากโปรแกรมออโตแคด เพื่อนำข้อมูลของแบบชิ้นงานไปใช้ประโยชน์ต่อไป

6. เพิ่มทักษะการใช้งานและควบคุมเครื่องกัดซีเอ็นซี

7. โปรแกรมที่ได้จากวิทยานิพนธ์เป็นโปรแกรมพื้นฐานที่ในการพัฒนาซอฟต์แวร์ CAM ต่อไปในอนาคต

บทที่ 2
ทฤษฎีที่เกี่ยวข้อง

2.1 เทคโนโลยีเอ็นซีและซีเอ็นซี

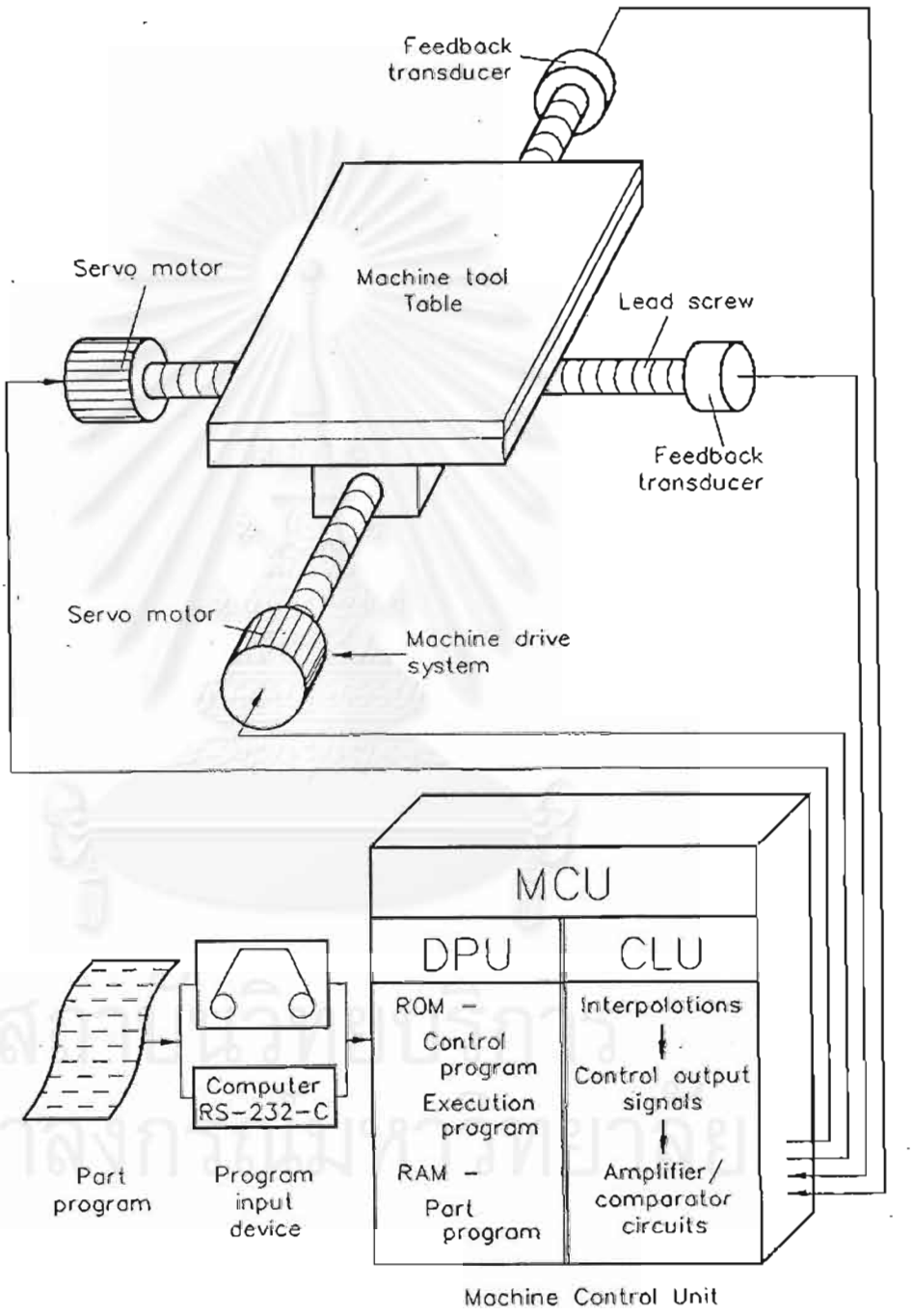
สมาคม Electronic Industries Association ให้คำนิยาม การควบคุมเชิงตัวเลข (Numerical Control, NC) ไว้ว่าเป็นระบบที่ควบคุมด้วยข้อมูลที่เป็นเชิงตัวเลขโดยตรงขณะทำงานอยู่ โดยระบบจะทำการแปลข้อมูลเชิงตัวเลขที่ได้รับมาเป็นสัญญาณควบคุม ซึ่งสัญญาณควบคุมที่ทำการแปลงได้นั้นจะมีอยู่ 2 ส่วนด้วยกันคือ สัญญาณอิเล็กทรอนิกส์ขาออกแบบพัลส์ (Pulses of Electrics output signal) กับสัญญาณควบคุมแบบเปิดปิด (On/Off Control signal) โดยสัญญาณอิเล็กทรอนิกส์ขาออกแบบพัลส์จะเป็นสัญญาณที่ใช้ในการควบคุมตำแหน่งแกนหมุน (Spindle) ของเครื่อง สัญญาณควบคุมแบบเปิดปิดจะเป็นสัญญาณที่ใช้ควบคุมความเร็วและทิศทางของการหมุนของแกนหมุน ควบคุมระบบหล่อเย็น และควบคุมการเลือกใช้หัวกัด (End mill)

ส่วนประกอบพื้นฐานหลักของระบบเอ็นซีและซีเอ็นซี ประกอบด้วย 6 ส่วน คือ

1. ส่วนการเขียนโปรแกรม (Part program)
2. เครื่องมือนำส่งโปรแกรม (Program input device)
3. หน่วยควบคุมเครื่องจักร (Machine control unit, MCU)
4. ระบบขับเคลื่อน (Drive system)
5. เครื่องมือกล (Machine tool)
6. ระบบป้อนกลับ (Feedback system)

ในส่วนส่วนการเขียนโปรแกรมก็คือการเขียนรหัสที่ใช้ในการควบคุมการเคลื่อนที่ของเครื่องมือกล เมื่อทำการป้อนโปรแกรมหรือรหัสเข้าสู่เครื่องเอ็นซีหรือซีเอ็นซี หน่วยควบคุมเครื่องจักรจะทำการการแปลงรหัสที่ได้รับให้เป็นสัญญาณทางไฟฟ้าเพื่อไปควบคุมส่วนต่างๆ ของเครื่องให้เป็นไปตามที่รหัสที่ป้อนเข้าไป โดยที่ หน่วยควบคุมเครื่องจักร จะอ่านและแปลงรหัสที่ป้อนเข้าไปทีละบรรทัด สัญญาณที่ได้จากการแปลงรหัสจะถูกจ่ายเข้าสู่เครื่องขยายเซอร์โว (Servo amplifiers) เพื่อที่จะขยายสัญญาณไปควบคุมกลไกเซอร์โว (Servo Mechanism) กลไกเซอร์โวอาจจะเป็นมอเตอร์ไฟฟ้าหรือระบบไฮดรอลิก (Hydraulic) ก็ได้ โดยมอเตอร์จะส่งถ่ายกำลังผ่านทางอุปกรณ์ที่มีชื่อว่า เกลียวนำแบบบอล (Ball lead screws) เพื่อที่จะควบคุมตำแหน่งของ Machine table หรือ จานหมุน และที่ปลายอีกด้านหนึ่งของเกลียวนำแบบบอล จะติดตั้งอุปกรณ์ส่งสัญญาณกลับ อุปกรณ์ส่งสัญญาณกลับนี้จะทำการตรวจวัดทิศทาง ตำแหน่ง และความเร็ว เป็นต้น แล้วส่งสัญญาณทางไฟฟ้าที่เป็นข้อมูลที่วัดได้กลับไปยังหน่วยควบคุมเครื่องจักร หน่วยควบคุมเครื่องจักรจะทำการเปรียบเทียบสัญญาณส่งกลับกับ

สัญญาณที่ได้จากการแปลงรหัส ถ้าเกิดผลต่างทางสัญญาณขึ้น หน่วยควบคุมเครื่องจักร ก็จะส่งสัญญาณควบคุมต่อไปจนกระทั่งไม่เกิดผลต่างทางสัญญาณเกิดขึ้น ดังได้แสดงพื้นฐานหลักในการทำงานอยู่ในรูปที่ 2.1



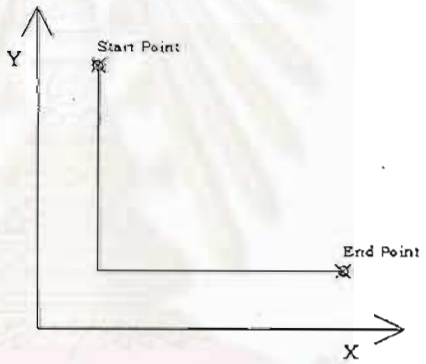
รูปที่ 2.1 รูปส่วนประกอบพื้นฐานหลักของเครื่องเอ็นซีและซีเอ็นซี

หอสมุดกลาง สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

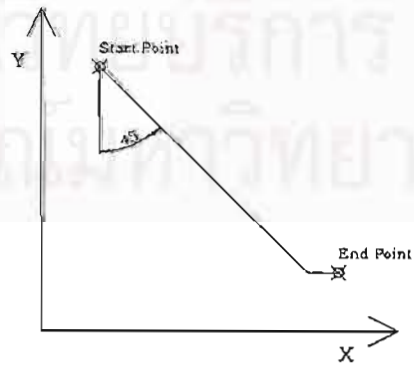
2.1.1 ระบบควบคุมของเครื่องเอ็นซีและซีเอ็นซี

2.1.1.1 ระบบควบคุมแบบจากจุดถึงจุด (Point to Point, PTP) ระบบควบคุมแบบจากจุดถึงจุดเป็นการควบคุมการเคลื่อนที่ของเครื่องมือกล จากจุดหนึ่งไปยังอีกจุดหนึ่งโดยไม่ถูกชิ้นงาน ซึ่งระบบควบคุมการเคลื่อนที่แบบนี้จะใช้ในงานเจาะ คว้าน เป็นต้น สามารถแบ่งตามลักษณะการเคลื่อนที่ของเครื่องมือกล ได้อีก 3 ประเภท คือ

1. ระบบควบคุมแบบเดินในแนวแกน (Axial Path) เครื่องมือกลจะเคลื่อนที่ทีละแกนและเคลื่อนที่ด้วยความเร็วสูงสุด ตามรูปที่ 2.2 จะเห็นว่าจะเครื่องมือกลจะเคลื่อนที่ในแนวแกน Y ก่อนด้วยความเร็วสูงสุดของเครื่องจักร เมื่อเครื่องมือกลเคลื่อนที่ถึงพิกัด Y ที่กำหนด เครื่องมือกลจะไม่เคลื่อนที่ในแนวแกน Y แต่จะเคลื่อนที่ในแนวแกน X แทนและจะเคลื่อนที่จนถึงค่าพิกัด X ที่กำหนด โดยใช้ความเร็วสูงสุดของเครื่อง ซึ่งข้อดีของระบบควบคุมแบบนี้ก็คือเป็นวิธีที่ควบคุมง่าย ไม่ต้องมีการคำนวณหาความเร็วในแต่ละแกน



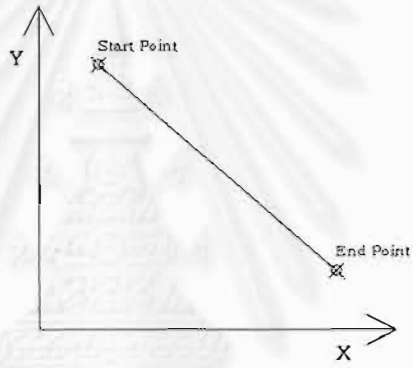
รูปที่ 2.2 แสดงลักษณะการเคลื่อนที่แบบเดินในแนวแกน



รูปที่ 2.3 แสดงลักษณะการเคลื่อนที่แบบเดินเป็นเส้นตรงทำมุม 45^{o}

2. ระบบควบคุมแบบเดินเป็นเส้นตรงทำมุม 45° (45° Line Path) ซึ่งแสดงตัวอย่างไว้ที่รูปที่ 2.3 โดยเครื่องมือกลจะเคลื่อนที่จากจุดเริ่มต้นเป็นมุม 45 ° ซึ่งหน่วยควบคุมเครื่องจักรจะคำนวณหาความเร็วในแต่ละแกนที่ทำให้เครื่องมือกลสามารถวิ่งทำมุม 45° ได้ เครื่องมือกลจะเคลื่อนที่จนถึงค่าพิกัด Y ก่อน ต่อจากนั้นก็จะวิ่งเป็นเส้นตรงขนานกับแกน X ละจะเคลื่อนที่โดยเคลื่อนที่ไปจนถึงค่าพิกัด X ที่กำหนด

3. ระบบควบคุมแบบเดินเชิงเส้น (Linear Path) ในการเคลื่อนที่ด้วยวิธีนี้ หน่วยควบคุมเครื่องจักรจะต้องมีความสามารถในการคำนวณหาเส้นทางที่สั้นที่สุดในการเคลื่อนที่ระหว่างจุด 2 จุด และจะต้องคำนวณหาความเร็วที่ควบคุมในแต่ละแกนที่ทำให้เครื่องมือกลเคลื่อนที่ไปถึงจุดปลายพร้อมๆ กันทุกแกน ตามรูปที่ 2.4 เป็นตัวอย่างระบบควบคุมแบบเดินเชิงเส้น



รูปที่ 2.4 แสดงลักษณะการเคลื่อนที่แบบเดินเชิงเส้น

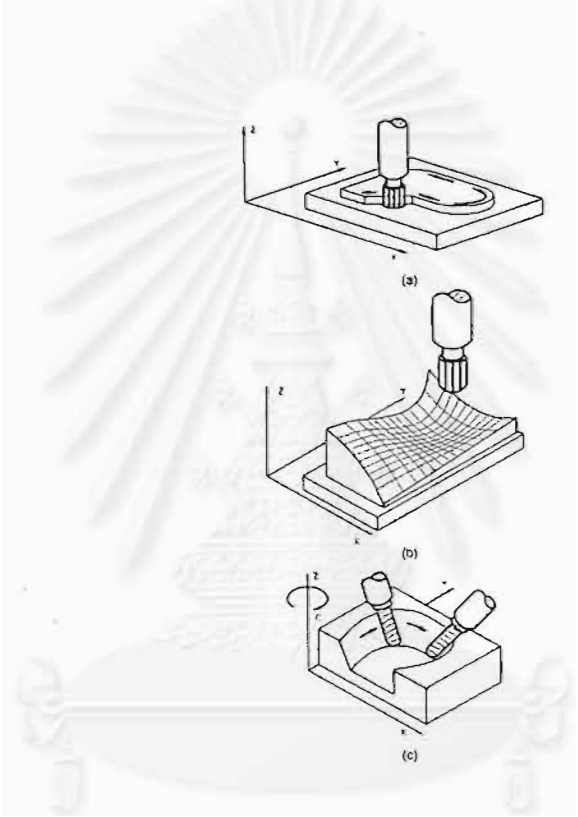
2.1.1.2 ระบบควบคุมแบบเดินต่อเนื่อง (Continuous Path) การควบคุมแบบเดินต่อเนื่องสามารถเรียกอีกชื่อหนึ่งว่า ระบบคอนทัวร์ (Contouring System) เป็นการควบคุมการเคลื่อนที่ของเครื่องมือกลให้เคลื่อนที่พร้อมๆ กันได้หลายแกน ตามทางเดิน (Path) ที่ถูกกำหนดขึ้น และสามารถควบคุมความเร็วในแต่ละแกนให้มีความเร็วที่แตกต่างกันได้ในขณะเวลาที่ควบคุมเดียวกันได้อย่างอิสระ การควบคุมชนิดนี้สามารถกัดชิ้นงานได้หลายรูปแบบอาทิเช่น การกัดแบบ 2 แกนครึ่ง การกัดแบบ 3 แกน การกัดแบบ 4 แกน เป็นต้น ซึ่งแสดงลักษณะการกัดที่ได้กล่าวมาไว้ที่รูปที่ 2.5 โดยการกัดแบบ 2 แกนครึ่งหัวกัดจะเคลื่อนที่กัดชิ้นงานเฉพาะแกน X และ Y เท่านั้น และแกน Z จะเคลื่อนที่ได้เมื่อหัวกัดไม่ได้กัดชิ้นงาน ซึ่งแสดงไว้ที่รูปที่ 2.5.a และในรูปที่ 2.5.b เป็นการกัดแบบ 3 แกนมีลักษณะการกัดชิ้นงานที่หัวกัดสามารถเคลื่อนที่กัดได้พร้อมๆ กันทั้ง 3 แกน ส่วนรูปที่ 2.5.c เป็นการกัดแบบ 4 แกน ซึ่งมีรูปแบบการกัดเมื่อแบบ 3 แกน แต่เพิ่มการวางทิศทาง (Orientation) รอบแกน Z ซึ่งจะเพิ่มความสามารถในการกัด

2.1.2 การประมาณค่าของเอ็นซีและซีเอ็นซี (NC and CNC Interpolation)

การประมาณค่า (Interpolation) สำหรับเครื่องเอ็นซีจะกระทำโดยใช้วงจรทางอิเล็กทรอนิกส์ และสำหรับเครื่องซีเอ็นซีจะเป็นตัวโปรแกรมที่ใช้ในการคำนวณ โดยตัวประมาณค่านี้จะมีหน้าที่อยู่ 2 อย่างด้วยกัน คือ

1. ช่วยในการคำนวณหาความเร็วที่ใช้ในการควบคุมการเคลื่อนที่ของเครื่องมือกลในแต่ละแกนให้เป็นไปตามที่โปรแกรมหรือรหัสที่ป้อนเข้ามาในเครื่อง

2. ช่วยในการคำนวณหาตำแหน่งของการเคลื่อนที่ของเครื่องมือกล



รูปที่ 2.5 แสดงลักษณะการเคลื่อนที่แบบระบบต่อเนื่อง

a) แบบ 2 แกนครึ่ง (Two-half Axis Contour Machining)

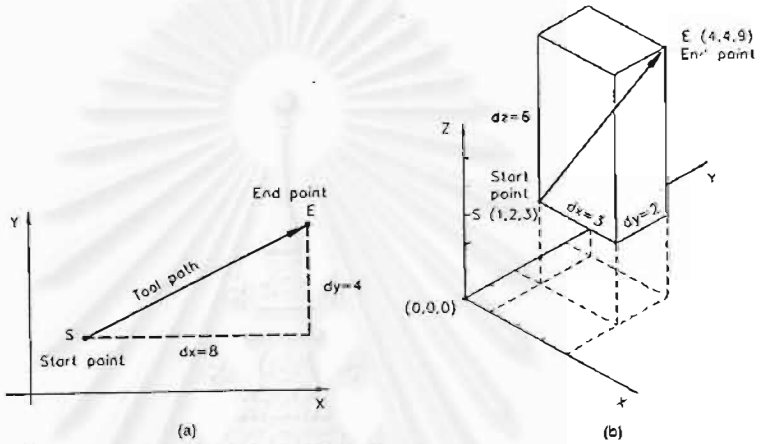
b) แบบ 3 แกน (Three Axis Contour Machining)

c) แบบ 4 แกน (Four Axis Contour Machining)

เทคนิคการประมาณค่าของเครื่องเอ็นซีและซีเอ็นซีสามารถจำแนกได้เป็น 5 วิธี

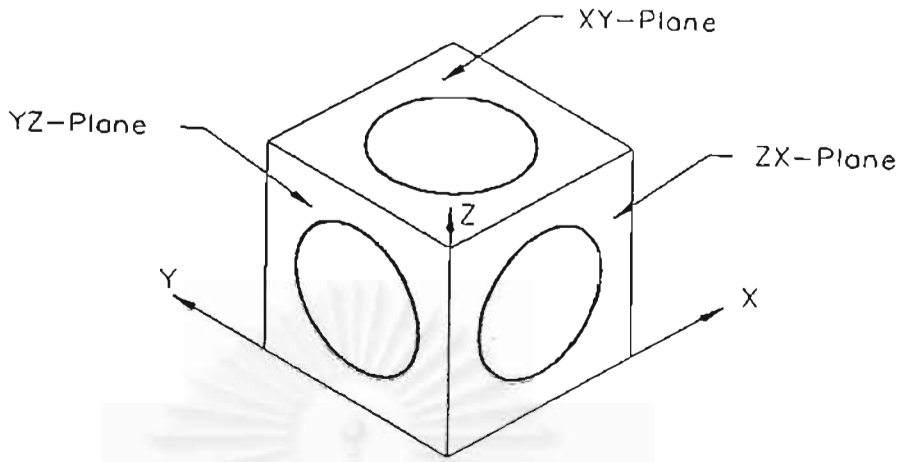
1. การประมาณค่าแบบเชิงเส้น (Linear Interpolation) เทคนิคนี้เครื่องมือกลจะเคลื่อนที่จากจุดหนึ่งไปยังอีกจุดหนึ่งเป็นเส้นตรง ในการเคลื่อนที่ของเครื่องมือกลที่เคลื่อนที่เป็นเส้นตรงต่อกันหลายๆ เส้น ตัวประมาณค่า (Interpolator) จะคำนวณหาเฉพาะจุดปลายของแต่ละเส้นเท่านั้น ก็เนื่องจากจุดปลายของเส้นที่ผ่านมาก็จะเป็นจุดเริ่มต้นของเส้นถัดไป ในการประมาณค่าแบบเชิงเส้นจะต้องการตัวพารามิเตอร์อยู่ 3 ตัวคือ 1. จุดเริ่มต้น

2. จุดปลาย 3. ความเร็วในแต่ละแกน โดยที่ตัวประมาณค่าจะคำนวณหาความเร็วในแต่ละแกนที่เหมาะสมให้เอง ในรูปที่ 2.6.a เป็นตัวอย่างการประมาณค่าแบบเชิงเส้นแบบ 2 แกน เมื่อตัวประมาณค่ารับค่าพิกัดจุเริ่มต้นและจุดปลายแล้ว ตัวประมาณค่าก็จะหาค่าผลต่างในแนวแกน X และ Y จากนั้นก็นำผลต่างทั้ง 2 ไปเทียบกับเวลาก็จะได้ความเร็วที่ใช้ในการควบคุม ส่วนในรูปที่ 2.6.b เป็นตัวอย่างการประมาณค่าแบบเชิงเส้นแบบ 3 แกน ซึ่งขั้นตอนการทำงานก็จะเหมือนกับแบบ 2 แกน แต่เพิ่มอีกหนึ่งแกนคือ แกน Z



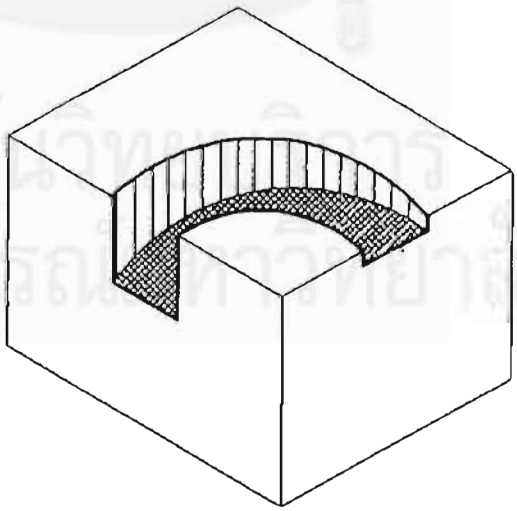
รูปที่ 2.6 แสดงลักษณะการประมาณค่าแบบเชิงเส้น
a) แบบ 2 แกน (Two Axis Linear Interpolation)
b) แบบ 3 แกน (Three Axis Linear Interpolation)

2. การประมาณค่าแบบเชิงวงกลม (Circular Interpolation) ในการประมาณค่าเทคนิคนี้วงกลมจะถูกแบ่งออกเป็นส่วนเล็กๆ ที่เป็นเส้นตรงโดยมีขนาดประมาณ 0.0025-0.005 มม. และมีการคำนวณหาความเร็วในการควบคุมเครื่องมือกลตามแกน X และ Y การประมาณค่าแบบเชิงวงกลมจะใช้ในการคำนวณได้เฉพาะงาน 2 แกน หรือระนาบ (Plane) เท่านั้น เช่น ระนาบ XY ระนาบ XZ และระนาบ YZ โดยแสดงรูปตัวอย่างการประมาณค่าแบบเชิงวงกลมในแต่ระนาบไว้ในรูปที่ 2.7 ในการคำนวณด้วยเทคนิคนี้จะต้องมีพารามิเตอร์ดังต่อไปนี้ 1) จุดเริ่มต้น 2) จุดปลาย 3) รัศมีหรือจุดศูนย์กลาง 4) ทิศทางการเคลื่อนที่



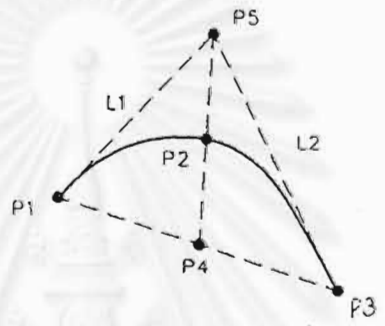
รูปที่ 2.7 การประมาณค่าแบบเชิงวงกลมในระนาบต่างๆ

3. การประมาณค่าแบบเชิงเกลียว (Helical Interpolation) ส่วนมากการประมาณค่าแบบเชิงเกลียวจะมีอยู่เฉพาะในเครื่องซีเอ็นซีรุ่นใหม่ๆ ซึ่งการประมาณค่าเทคนิคนี้จะเป็นการรวมเอาการประมาณค่าแบบเชิงเส้นกับการประมาณค่าแบบเชิงวงกลมมารวมเข้าไว้ด้วยกัน โดยความสามารถการประมาณค่าแบบนี้สามารถคำนวณหาเส้นทางเดินของหัวกัด (Tool Path) ที่เป็นลักษณะ 3 แกนได้ โดยแสดงรูปตัวอย่างไว้ในรูปที่ 2.8 ซึ่งในรูปตัวอย่างนี้จะคำนวณหาค่าพิกัด X และ Y ด้วยการประมาณค่าแบบเชิงวงกลม แต่ค่าพิกัด Z จะใช้การประมาณค่าแบบเชิงเส้น



รูปที่ 2.8 แสดงลักษณะการประมาณค่าแบบเชิงเกลียว

4. การประมาณค่าแบบเชิงพาราโบลา (Parabolic Interpolation) จะเริ่มต้นการคำนวณหาเส้นทางเดินของหัวกัดจากจุด 3 จุดที่ไม่ได้อยู่บนเส้นตรงเดียวกัน เพื่อที่จะนำมาคำนวณหาเส้นโค้ง โดยแสดงตามรูปที่ 2.9 จุด P1 จุด P2 และจุด P3 เป็นจุดที่กำหนดให้เส้นโค้งผ่านทั้ง 3 จุด และ จุด P2 อยู่กึ่งกลางระหว่างจุด P4 และจุด P5 จุด P4 อยู่กึ่งกลางระหว่างจุด P1 และจุด P3 จากนั้นก็จะได้เส้นตรง L1 และ L2 มาช่วยในการสร้างเส้นโค้ง ซึ่งเส้นโค้งจะต้อง สัมผัส (Tangent) กับเส้นตรง L1 และ L2 ตรงจุด P1 และ P3 ตามลำดับ



รูปที่ 2.9 แสดงลักษณะการประมาณค่าแบบเชิงพาราโบลา

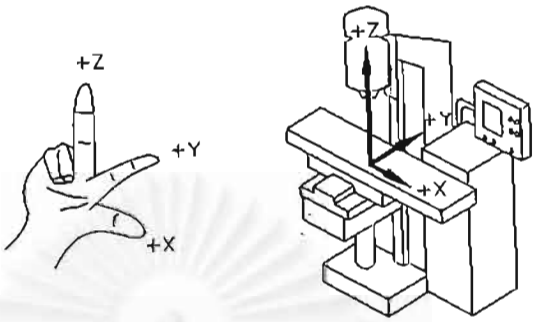
5. การประมาณค่าแบบยกกำลัง 3 (Cubic Interpolation) จะใช้ในงานคำนวณหาเส้นทางเดินของหัวกัดของชิ้นงานที่รูปแบบยากๆ เช่น งาน Automobile sheet metal dies เป็นต้น ซึ่งเทคนิคนี้ยังมีความสามารถไม่เพียงแค่คำนวณหาเส้นทางเดินของหัวกัดที่เป็นเส้นโค้ง แต่ยังสามารถคำนวณหาเส้นทางเดินของหัวกัดของเส้นโค้งหลายเส้นที่ต่อเชื่อมกันได้อย่างราบเรียบกัน

2.1.3 ระบบพิกัด (Coordinate System)

ในการเขียนรหัสควบคุมเครื่องเอ็นซีและซีเอ็นซีให้กัดชิ้นงานตามที่ต้องการนั้น จำเป็นจะต้องมีการวางแผนก่อนว่า จะต้องใช้คำสั่งใดในการควบคุม การเลือกใช้เครื่องมือกล ชนิดไหนจึงจะเหมาะสมกับงาน เครื่องเอ็นซีหรือซีเอ็นซีแบบใดที่จะใช้งาน สิ่งเหล่านี้เป็นสิ่งที่ต้องรู้และต้องมีการวางแผนก่อน ในการแปลงสิ่งเหล่านี้ให้เป็นรูปแบบคำสั่งที่ใช้ในการควบคุมจำเป็นที่จะต้องใช้ระบบพิกัดเพื่อที่จะใช้ในการบอกตำแหน่งที่ต้องการ ระบบพิกัดที่ใช้งานในปัจจุบันแบ่งออกเป็น 2 ประเภท

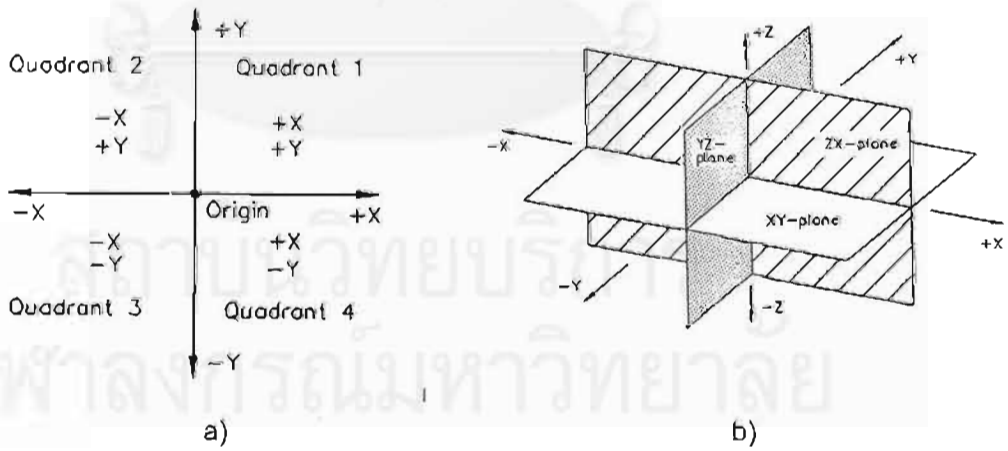
2.1.3.1 ระบบพิกัดฉาก (Cartesian Coordinate System หรือ Rectangular Coordinate System) การเคลื่อนที่ของเครื่องมือกลของเครื่องเอ็นซีและซีเอ็นซี จะเป็นแบบระบบพิกัดฉาก เนื่องจากโครงสร้างของเครื่องเอ็นซีและซีเอ็นซีส่วนใหญ่จะสร้างบนพื้น

ฐานที่เป็นแกนพิกัดฉาก 3 แกน รูปที่ 2.10 แสดงถึงเครื่องกัดซีเอ็นซีแบบตั้งในระบบพิกัดฉาก ซึ่งมีทิศทางของหัวกัดใน 3 ทิศทางตั้งฉากกัน โดยสามารถแสดงทิศทางด้วยกฎมือขวา



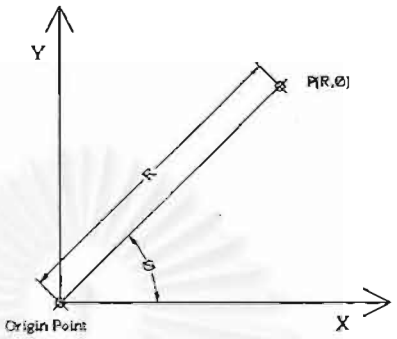
รูปที่ 2.10 เครื่องกัดซีเอ็นซีแบบตั้งในระบบพิกัดฉาก

แต่ในระบบ 2 แกนที่เป็นระบบพิกัดฉาก อาทิ ระนาบ XY จะประกอบไปด้วย 2 แกนที่ตัดตั้งฉากกัน ดังแสดงไว้ในรูปที่ 2.11.a ซึ่งแสดงระบบพิกัดฉากแบบ 2 แกน ซึ่งจุดกำเนิดอยู่ที่จุดตัดระหว่างแกน X และแกน Y และในระบบ 3 แกน ก็จะเป็นการตัดกันและตั้งฉากทั้ง 3 แกน คือ แกน X แกน Y และแกน Z และเรียกจุดตัดเป็นจุดกำเนิดเช่นกัน โดยแสดงไว้ในรูปที่ 2.11.b และในการพิจารณาทิศทางใดในแต่ละแกนมีค่าเป็นบวกหรือลบนั้น จะใช้วิธีกฎมือขวา



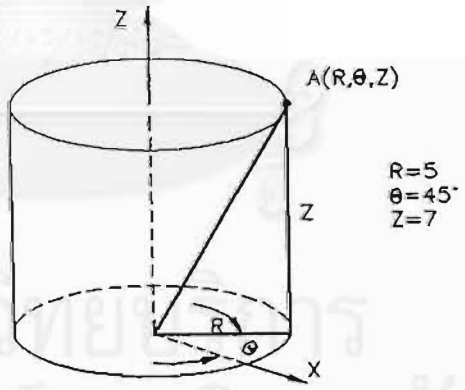
รูปที่ 2.11 ระบบพิกัดฉาก a) แบบ 2 แกน b) แบบ 3 แกน

2.1.3.2 ระบบพิกัดเชิงขั้ว (Polar Coordinate System) ของระบบ 2 แกน หรือระนาบ XY เป็นการกำหนดค่าตำแหน่งจากค่ารัศมี โดยวัดจากจุดอ้างอิงหรือจุดกำเนิดเป็นมุมขึ้นไปซึ่งเทียบกับแกนอ้างอิงหรือแกน X ตามรูปที่ 2.12



รูปที่ 2.12 ระบบพิกัดเชิงขั้วแบบ 2 แกน

ถ้าเพิ่มแกนเข้าไปในระบบอีกหนึ่งแกนนั่นก็คือ แกน Z ระบบพิกัดเชิงขั้วก็จะถูกเปลี่ยนเป็นระบบพิกัดทรงกระบอก (Cylindrical Coordinate System) ซึ่งประกอบไปด้วย รัศมี (R) มุม (θ) และ แกน Z ซึ่งแสดงตามรูปที่ 2.13



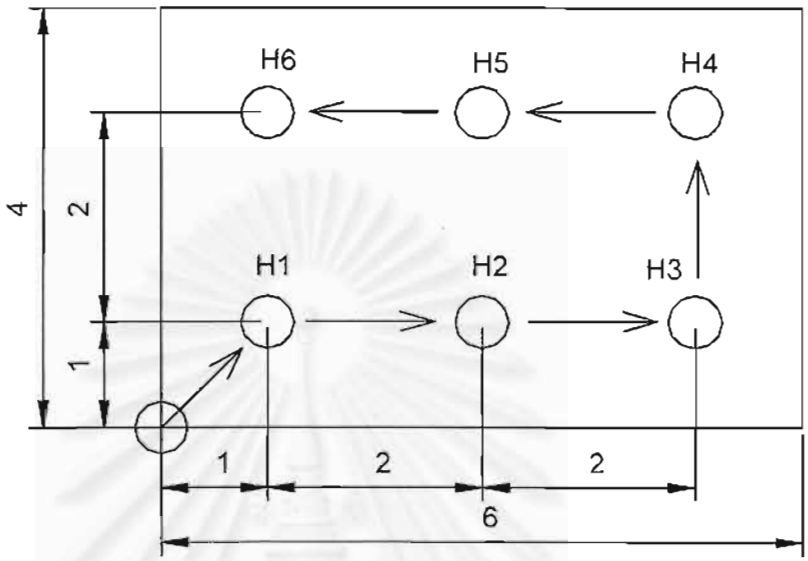
รูปที่ 2.13 ระบบพิกัดเชิงขั้วแบบ 3 แกนหรือระบบพิกัดทรงกระบอก

2.1.4 ระบบตำแหน่ง (Positioning System)

ในส่วนของการโปรแกรมรหัสที่ใช้ในการควบคุมเครื่องเอ็นซีและซีเอ็นซีให้ เครื่องมือกล เคลื่อนที่มีลักษณะการป้อนค่าตัวเลขที่ควบคุมการเคลื่อนที่อยู่ 2 ประเภท คือ

2.1.4.1 ระบบตำแหน่งแบบสัมบูรณ์ (Absolute Positioning System) ตำแหน่งในระบบตำแหน่งแบบสัมบูรณ์ทุกตำแหน่งจะถูกวัดจากจุดกำเนิดเดียวของรูปชิ้นงาน โดยจุดกำเนิดนี้จะถูกกำหนดตามตำแหน่งที่ผู้โปรแกรมพอใจหรือคิดว่าเป็นจุดที่ดีที่สุด จากรูปที่

2.14 เป็นรูปที่แสดงแบบชิ้นงานในการเจาะรู โดยกำหนดให้จุดกำเนิดอยู่ที่มุมซ้ายด้านล่าง และต้องการเจาะจากรู H1 ถึงรู H6 ตามลำดับ ซึ่งตำแหน่งของรูในระบบตำแหน่งแบบสัมบูรณ์ที่เครื่องมือกลจะเคลื่อนที่ไปเจาะรูได้แสดงไว้ในตารางที่ 2.1



รูปที่ 2.14 ภาพการเจาะรูจำนวน 6 รู จาก H1 ถึง H6 ตามลำดับ

Sequence	X Coordinate	Y Coordinate
H1	1.0	1.0
H2	3.0	1.0
H3	5.0	1.0
H4	5.0	3.0
H5	3.0	3.0
H6	1.0	3.0

ตารางที่ 2.1 แสดงค่าพิกัด XY ด้วยระบบตำแหน่งแบบสัมบูรณ์

2.1.4.2 ระบบตำแหน่งแบบส่วนเพิ่ม (Incremental Positioning System) ระบบนี้จุดกำเนิดจะไม่ถูกกำหนดตายตัว แต่จะเคลื่อนที่จากจุดหนึ่งไปยังอีกจุดหนึ่ง ระยะที่เครื่องมือกลเคลื่อนที่จะคำนวณจากระยะทางระหว่างตำแหน่งของเครื่องมือกลในปัจจุบันกับตำแหน่งของเครื่องมือกลในตำแหน่งต่อไป โดยเครื่องหมายบวกและลบจะแสดงถึงทิศทางในการเคลื่อนที่ของเครื่องมือกล ดังแสดงเป็นตัวเลขในตารางที่ 2.2 โดยใช้รูปที่ 2.14 มาเป็นรูปต้นแบบ

Sequence	X Coordinate	Y Coordinate
H1	1.0	1.0
H2	2.0	0.0
H3	2.0	0.0
H4	0.0	2.0
H5	-2.0	0.0
H6	-2.0	0.0

ตารางที่ 2.2 แสดงค่าพิกัด XY ด้วยระบบตำแหน่งแบบส่วนเพิ่ม

2.1.5 พิกัดอ้างอิง (Reference Coordinate)

ในการโปรแกรมรหัสควบคุมจำเป็นจะต้องมีจุดอ้างอิง (Reference Points) อย่างน้อยหนึ่งจุด ที่จะใช้ในการคำนวณหาจุดต่างๆ บนชิ้นที่ต้องการ ซึ่งสามารถจำแนกได้ดังต่อไปนี้

1. จุดอ้างอิงของเครื่องจักร (Machine Reference Point)

จุดนี้เป็นจุดกำเนิดของเครื่องหรือเรียกว่า Machine home reference ซึ่งจะถูกกำหนดมาจากบริษัทที่ผลิตเครื่องเอ็นซีหรือซีเอ็นซี และไม่สามารถเปลี่ยนแปลงจุดนี้ได้

2. จุดอ้างอิงของโปรแกรม (Program Reference Point)

จุดนี้จะถูกกำหนดโดยผู้เขียนแบบรูปที่ต้องการหรือจากผู้ที่เขียนโปรแกรมรหัสควบคุม โดยจุดนี้จะเป็นจุดกำเนิดของแบบหรือภาพต้นแบบที่ต้องการ โดยจุดนี้จะใช้เป็นจุดอ้างอิงในการคำนวณจุดต่างๆ ในการเคลื่อนที่ของ เครื่องมือกล

3. จุดอ้างอิงของงาน (Work Reference Point)

ในการกัดหรือเจาะชิ้นงาน 1 แบบบนเครื่องซีเอ็นซี สามารถกัดหรือเจาะชิ้นงานได้หลายชิ้น โดยที่ติดตั้งชิ้นงานเพียงครั้งเดียว โดยแต่ละชิ้นงานจะต้องมีจุดอ้างอิงของแต่ละชิ้นและจะต้องสัมพันธ์กับจุดอ้างอิงของโปรแกรม ด้วย ซึ่งจุดอ้างอิงนี้ก็คือจุดอ้างอิงของงาน

2.1.6 คำสั่งที่ใช้สำหรับเครื่องกัด เอ็นซี และ ซีเอ็นซี

ในการกัดชิ้นงานด้วยการควบคุมด้วยตัวเลขสามารถที่จะใช้เครื่องกัดเอ็นซี และ ซีเอ็นซี หรือ เครื่องแมชชีนนิ่ง เซนเตอร์ (Machining Center) ก็ได้ การควบคุมเครื่องเหล่านี้จำเป็นต้องเขียนโปรแกรมหรือรหัสที่มีลักษณะเพื่อควบคุมสั่งงาน โดยรูปแบบรหัสก็จะต้องใช้ให้เหมาะสมกับเครื่องแต่ละชนิดและแต่ละผู้ผลิต ซึ่งจะมีรูปแบบที่แตกต่างกันเล็กน้อย การเขียนโปรแกรมหรือรหัสอาจจะกระทำได้หลายวิธี เช่น การเขียนด้วยคนโดยตรง

การใช้โปรแกรมช่วยในการเขียน เป็นต้น โปรแกรมหรือรหัสที่ใช้ควบคุมจะประกอบไปด้วยตัวอักษรหรือรหัส กับ ค่าตัวเลข โดยกลุ่มของตัวอักษรหรือรหัสจะแสดงดังตารางที่ 2.3

ฟังก์ชัน	รหัส	ความหมาย
หมายเลขลำดับที่	N	หมายเลขลำดับที่ของโปรแกรม
หมายเลขการเตรียมตัว	G	กำหนดโหมดการทำงานเส้นตรง เส้นโค้ง เป็นต้น
ค่าพิกัด	XYZ	ค่าตำแหน่งหรือระยะทางในแกนพิกัดฉาก
	IJK	ค่าพิกัดของศูนย์กลางส่วนโค้ง
ความเร็วป้อน	F	กำหนดค่าความเร็วป้อน
ฟังก์ชันแกนหลัก	S	กำหนดค่าจำนวนการหมุนของแกนหลัก
ฟังก์ชันหัวกัด	T	กำหนดหมายเลขของหัวกัด
ฟังก์ชันย่อย	M	กำหนดการควบคุมแบบเปิดปิด

ตารางที่ 2.3 แสดงความหมายของกลุ่มรหัส

รหัสที่ใช้ในการสั่งงานของโปรแกรมสามารถแบ่งกลุ่มรหัสได้ 5 กลุ่ม คือ

1. รหัสคำสั่งพื้นฐาน
   - รหัสควบคุมการเคลื่อนที่ (G00, G01, G02 และG03)
   - รหัสการเลือกระนาบ (G17, G18 และG19)
   - รหัสการเลือกระบบตำแหน่ง (G90 และG91)
   - รหัสการเลือกหน่วย (G70 หรือ G20, G71 หรือ G21)
   - รหัสกำหนดพิกัดงาน (G92)
   - รหัสเลือกหัวกัด และเปลี่ยนหัวกัด (Txx และG06)
   - รหัสกำหนดอัตราป้อน (Fxxxx, G94 และG95)
   - รหัสกำหนดอัตราหมุนของแกนหมุนและควบคุม (Sxxx, M03, M04)
   - รหัสฟังก์ชันเบ็ดเตล็ด (M00-M02, M07-M09 และM30)
2. รหัสคำสั่งการชดเชย (Compensation) และ ออฟเซต (Offset)
   - รหัสชดเชยพิกัดงาน (G54-G59)
   - รหัสชดเชยเส้นผ่าศูนย์กลางหัวกัด (G40-G42)
   - รหัสออฟเซตความยาวหัวกัด (G43, G44 และG49)

3. รหัสคำสั่งวัฏจักรคงที่ Fixed Cycle
   - รหัสกำหนดวัฏจักรคงที่แบบมาตรฐาน (G80-G89)
   - รหัสกำหนดวัฏจักรคงที่แบบพิเศษ (G73, G74 และG76)
   - รหัสกำหนดวัฏจักรที่ผู้ใช้กำหนด
4. มาโครและโปรแกรมแบบซับรูทีน (Macro and Subroutine Programming)

ฟังก์ชันนี้จะมีอยู่ในเฉพาะเครื่องซีเอ็นซีรุ่นใหม่ๆ จะมีคุณสมบัติในการกำหนดตัวแปร เพื่อที่จะใช้ในการคำนวณและตัดสินใจที่เป็นแบบตรรกศาสตร์ (Logic) ซึ่งคุณสมบัติเหล่านี้จะทำให้เครื่องซีเอ็นซีมีความสามารถในการกระทำซ้ำๆ ได้

5. รูปแบบการโปรแกรมแบบชั้นสูง (Advanced Programming Features)

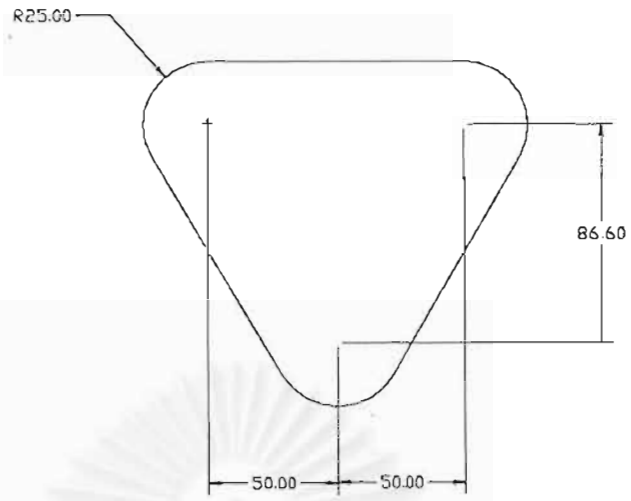
ในส่วนนี้จะเป็นรูปแบบในการเขียนโปรแกรมหรือรหัสควบคุมเครื่องเอ็นซีและซีเอ็นซีที่สามารถช่วยลดเวลาในการเขียนโปรแกรมและช่วยลดขนาดของโปรแกรมด้วย

2.2 พื้นฐานของคอมพิวเตอร์ช่วยในการออกแบบและช่วยในการผลิต

ในกระบวนการออกแบบผลิตภัณฑ์ได้มีการเปลี่ยนแปลงและพัฒนาเป็นอย่างมากโดยมี เทคโนโลยีหนึ่งที่ชื่อว่าคอมพิวเตอร์ช่วยในการออกแบบ (Computer Aided Design, CAD) และคอมพิวเตอร์ช่วยในการการผลิต (Computer Aided Manufacturing, CAM) เกิดขึ้นและได้มาปฏิวัติวิธีการและการดำเนินการที่เกี่ยวกับการออกแบบชิ้นส่วนทางด้านเครื่องกลไฟฟ้า และวิธีการในการผลิต โดยเทคโนโลยีนี้ได้นำเอาระบบคอมพิวเตอร์เข้ามาช่วยในกระบวนการออกแบบตลอดจนกระบวนการผลิต ในส่วนของ CAD เป็นเทคโนโลยีของการสร้างโปรแกรมควบคู่กับเทคโนโลยีของฮาร์ดแวร์ประกอบกัน เพื่อที่จะสร้างภาพทั้งในลักษณะ 2 มิติ 3 มิติ และ การนำเสนอภาพออกแบบที่เสียนแบบของจริง CAD สามารถจำแนกตามวิธีการขึ้นรูปแบบจำลอง (Model) ได้ดังนี้

1. แบบจำลองโครงลวด (Wire Frame Model)

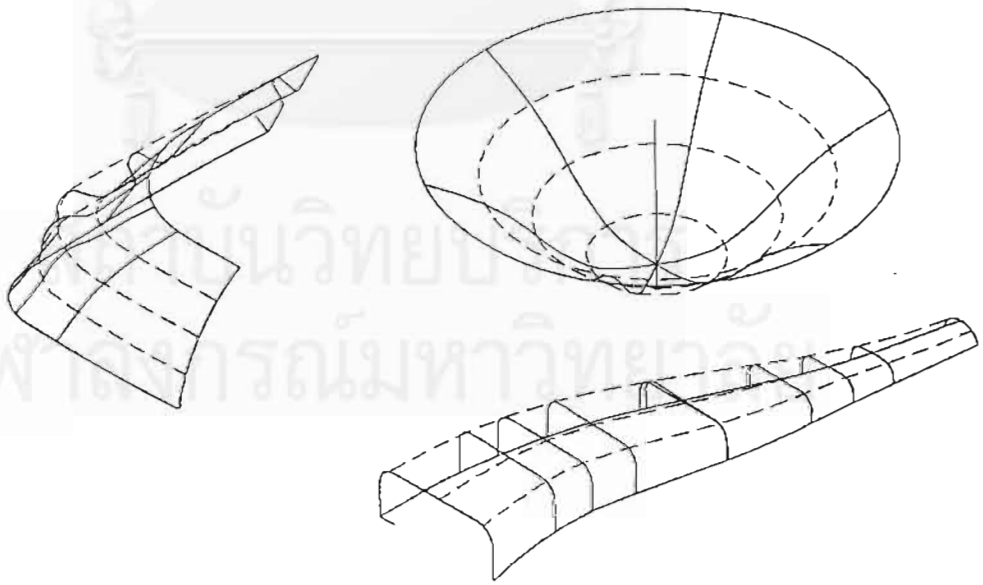
แบบจำลองโครงลวดเป็นแบบจำลองที่ถูกนำเสนอด้วยเส้นตรง เส้นโค้งวงกลม หรือ เส้นสไปน์ (Spline) เป็นต้น นำมาวาดเป็นเส้นโครงลวดตามแบบที่ต้องการ โดยแบบจำลองโครงลวด จะไม่มีผิวหน้าของแบบที่วาดขึ้น มีแต่เพียงเส้นขอบของแบบ ซึ่งสามารถวาดได้ทั้ง 2 มิติ และ 3 มิติ การสร้างภาพแบบจำลองโครงลวดจะค่อนข้างง่ายและไม่ซับซ้อน ประโยชน์ของแบบจำลองโครงลวด ก็ค่อนข้างจำกัด แต่ก็ยังสามารถนำมาใช้ช่วยในงานผลิตได้ เช่น งานไวร์ อีดีเอ็ม (Wire EDM) เป็นต้น ดังรูปที่ 2.15 แสดงตัวอย่างแบบชิ้นงานที่วาดขึ้นด้วยแบบจำลองโครงลวด



รูปที่ 2.15 ภาพแสดงรูปภาพที่วาดขึ้นจากแบบจำลองโครงลวด

2. แบบจำลองรูปผิว (Surface Model)

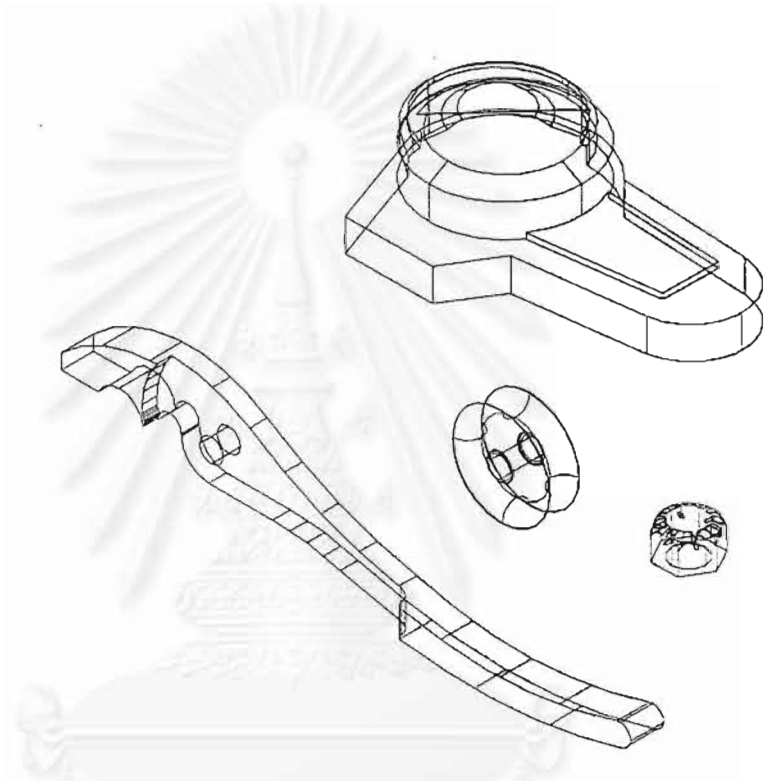
แบบจำลองรูปผิวจะมีรายละเอียดที่สมบูรณ์เหมือนวัตถุจริงมากกว่าแบบจำลองโครงลวด และก็จะมีความซับซ้อนมากขึ้น เนื่องจากจะมีการสร้างเส้นขอบและผิวของวัตถุที่วาดขึ้น ในเมื่อแบบจำลองแบบนี้มีผิวก็จะสามารถทำการลงสีและให้แสงเงาได้ (Shading) เพื่อตรวจสอบดูชิ้นงานได้ง่ายขึ้น ซึ่งได้แสดงแบบชิ้นงานตัวอย่างที่วาดจากแบบจำลองรูปผิวดังรูปที่ 2.16



รูปที่ 2.16 ภาพแสดงรูปภาพที่วาดขึ้นจากแบบจำลองรูปผิว

3. แบบจำลองโซลิด (Solid Model)

แบบจำลองโซลิดจะมีรายละเอียดเพิ่มขึ้นจากแบบจำลองรูปผิว โดยแบบจำลองโซลิด ไม่เพียงจะมีเส้นขอบและผิววัตถุแล้ว แต่ยังมีเนื้อในวัตถุอีกด้วย ดังนั้นแบบจำลองโซลิดจะเป็นแบบจำลองที่มีคุณสมบัติทึบและตัน ซึ่งคุณสมบัตินี้สามารถนำมาช่วยในการคำนวณหาค่าทางวิศวกรรมได้โดยตรง เช่น มวล (Mass) ค่าความหนาแน่น (Density) ค่าโมเมนความเฉื่อย (Moment of Inertia) เป็นต้น ดังรูปที่ 2.17 แสดงตัวอย่างแบบจำลองโซลิด



รูปที่ 2.17 ภาพแสดงรูปภาพที่วาดขึ้นจากแบบจำลองโซลิด

ในส่วนของ CAM เป็นซอฟต์แวร์ที่เกิดจากการพัฒนาเครื่องจักร เครื่องมือ จากการใช้แรงงานคนมาควบคุมให้เครื่องจักรทำงานกันอย่างอัตโนมัติ ดังนั้นในการควบคุมเครื่องจักรจำเป็นใช้รหัสในการควบคุม CAM จะเข้ามาช่วยในการแปลงข้อมูลจาก CAD มาเป็นรหัสที่ใช้ในการควบคุมเครื่องจักร เครื่องจักรที่ถูกควบคุมจะเป็นจำพวกเครื่องเอ็นซี (NC Machine) และเครื่องซีเอ็นซี (CNC Machine) CAM สามารถแบ่งตามการประยุกต์ใช้งานหรือตามประเภทของเครื่องจักรที่ใช้งาน ได้แก่ งานกลึง (Turning) งานกัด (Milling) งานไวร์คัท (Wire Cut) เป็นต้น CAM บางผู้ผลิตไม่ได้มีการพัฒนา CAD มาด้วยกัน ดังนั้นจะต้องรับข้อมูลจาก CAD ของผู้ผลิตอื่น ด้วยเหตุนี้จึงทำให้เกิดปัญหาในการแลกเปลี่ยนข้อมูลกันระหว่าง CAD กับ CAM ดังนั้นจึงได้มีการสร้างมาตรฐานการแลกเปลี่ยนข้อมูลขึ้น โดยมีมาตรฐานดังต่อไปนี้ IGES (Initial Graphics Exchange Specification) PDDI (Product Definition Data Interface)

SET (Standard d'E Change dt de Transfert) DXF (Drawing Exchange Format) เป็นต้น แต่ก็มี CAD/CAM ที่สร้างจากผู้ผลิตเดียวกันก็จะไม่เกิดปัญหาในการแลกเปลี่ยนข้อมูล

2.3 โปรแกรมออโตแคด (AutoCAD)

ในปัจจุบันมีบริษัทผู้ผลิตซอฟต์แวร์หลายบริษัทได้พัฒนา CAD บนเครื่องคอมพิวเตอร์ส่วนบุคคล และหนึ่งในบริษัทเหล่านั้นก็คือ บริษัทออโต้เดสก์ (Autodesk) ผู้ผลิตโปรแกรมออโตแคด โปรแกรมออโตแคดเป็นซอฟต์แวร์ CAD เท่านั้น ดังนั้น AutoCAD จึงถูกนำมาใช้งานช่วยในการออกแบบผลิตภัณฑ์ โปรแกรมออโตแคดได้ถูกพัฒนามาจนถึงออโตแคด 2000 โดยโปรแกรมออโตแคดรุ่นแรกๆ มีความสามารถเพียงในงานออกแบบงาน 2 มิติเท่านั้น ดังนั้นจึงทำให้โครงสร้างหลักของโปรแกรมออโตแคดอยู่กับการออกแบบ 2 มิติ แต่ในปัจจุบันโปรแกรมออโตแคดถูกพัฒนาในส่วนการออกแบบ 3 มิติเพิ่มเติมเข้าไป แต่ก็ยังไม่ดีเท่าที่ควรก็เนื่องด้วยโครงสร้างหลักยังเป็น 2 มิติอยู่ ดังนั้นบริษัทออโต้เดสก์จึงได้พัฒนาโปรแกรมที่ชื่อว่า แมคแคนนิคอล เดสก์ทอฟ (Mechanical Desktop) มาช่วยเพิ่มประสิทธิภาพในส่วนการออกแบบงาน 3 มิติของโปรแกรมออโตแคดให้มีความสมบูรณ์ยิ่งขึ้น โดยจะมาช่วยแก้ไขให้ส่วนของการสร้างแบบจำลองรูปผิว และการแก้ไขแบบจำลองรูปผิว เป็นต้น

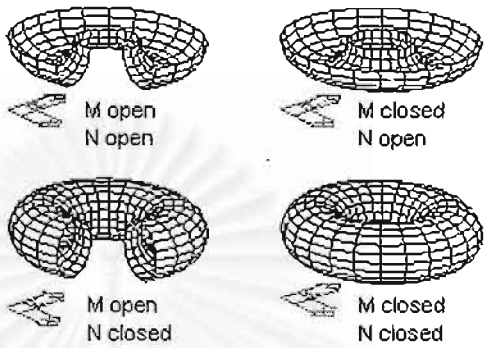
2.4 ตัวต่อประสาน ARX (ARX Interface)

ในการพัฒนาโปรแกรมที่ต้องการใช้งานร่วมกับโปรแกรมออโตแคดจำเป็นจะต้องศึกษาตัวต่อประสาน ที่ชื่อว่า ARX (AutoCAD Runtime Extension) ตัวต่อประสาน ARX จะช่วยในการเชื่อมติดต่อโปรแกรมที่พัฒนาให้สามารถทำงานร่วมกับโปรแกรมออโตแคด และสามารถเข้าถึงฐานข้อมูลของแบบชิ้นงานที่วาดขึ้นจากโปรแกรมออโตแคด ARX เป็น ไดนามิค ลิงค์ ไลบรารี (Dynamic Link Library, DLL) ที่ใช้ตำแหน่งที่ว่างของหน่วยความจำของโปรแกรมออโตแคด และยังสามารถเรียกใช้และสร้างฟังก์ชันใหม่ที่ทำงานบนโปรแกรมออโตแคดได้โดยตรง ในการเขียนโปรแกรม ARX จะเขียนด้วยหลักการออบเจ็กต์ โอเรียนเท็ด (Object-Oriented) ในภาษา C++ ARX เต็มไปด้วบเครื่องมือที่ใช้ได้อย่างมีประสิทธิภาพที่ใช้สำหรับจัดการระบบการจัดการกราฟฟิก การจัดการและเข้าถึงข้อมูลภายใน และการจัดการคำสั่งต่างๆ ของโปรแกรมออโตแคด ARX ยังถูกออกแบบให้สามารถทำงานร่วมกับ Visual LISP และชุดไลบรารี (Libraries) อื่นๆ ที่ผู้พัฒนาซอฟต์แวร์จะนำมาใช้ร่วมกันในการพัฒนาโปรแกรม

2.5 การวาดรูปผิวด้วยโปรแกรมออโตแคด

รูปผิวที่วาดขึ้นจากโปรแกรมออโตแคดถูกกำหนดให้แสดงด้วยโพลีกอนเมช ก็เนื่องด้วยผิวหน้าของแต่ละตาข่ายเป็นระนาบเรียบเล็กๆ เมื่อมีเมชหลายๆ เมชต่อกันก็จะได้รูปผิว ความละเอียดของรูปผิวจะถูกเก็บอยู่ในรูปเมทริกซ์ของเส้น M และ N ซึ่งคล้ายกับกริด

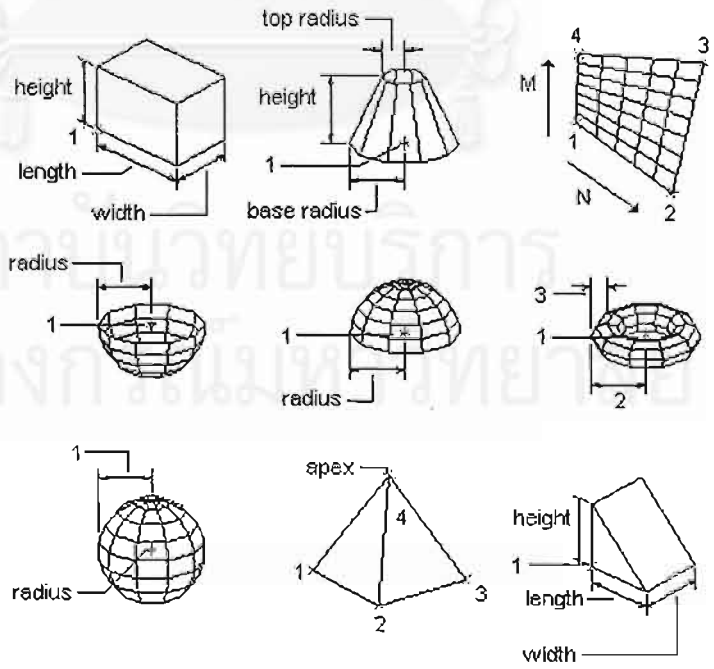
(Grid) ที่ประกอบด้วยคอลัมน์ (Columns) และแถว (Rows) โดยกำหนดให้เส้น M และ N เป็นคอลัมน์และแถว ตามลำดับ โพลีกอนเมชจำแนกได้เป็น โพลีกอนเมชปิด กับ โพลีกอนเมชเปิด โพลีกอนเมชที่จะปิดหรือเปิดจะต้องกำหนดด้วยเส้น M และ N ดังแสดงในรูปที่ 2.18



รูปที่ 2.18 ภาพแสดงโพลีกอนเมชเปิดและปิด

โปรแกรมออโตแคดมีรูปแบบการสร้างรูปผิวหลายรูปแบบ โดยผ่านคำสั่งดังต่อไปนี้

1. คำสั่ง 3D เป็นการสร้างรูปผิวดังต่อไปนี้ Boxes, Cones, Meshes, Dishes, Domes, Tori (Donuts), Spheres, Pyramids และ Wedges ซึ่งแสดงตัวอย่างรูปผิวไว้ที่รูปที่ 2.19



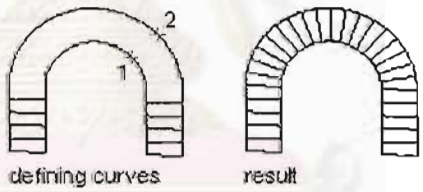
รูปที่ 2.19 ภาพแสดงรูปผิวที่สร้างจากคำสั่ง 3D

หอสมุดกลาง สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

2. คำสั่ง 3DMesh จะสร้างรูปผิวที่เป็นโพลีกอนเมชแบบเปิดทั้งเส้น M และ N รูปที่ 2.20 แสดงตัวอย่างแบบชิ้นงานที่วาดขึ้นจากคำสั่ง 3DMesh

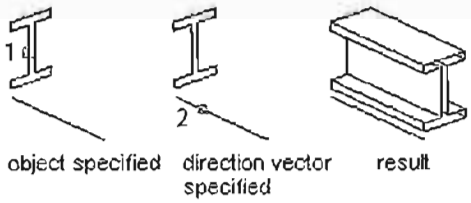
รูปที่ 2.20 ภาพแสดงความแตกต่างของค่า M และ N ของคำสั่ง 3DMesh

3. คำสั่ง RuleSurf เป็นการสร้างรูปผิว Ruled ซึ่งจะสร้างรูปผิวระหว่างเส้น 2 เส้น ซึ่งเส้นทั้ง 2 สามารถต่างชนิดกันได้ โดยที่ชนิดของเส้นที่สามารถใช้สร้างรูปผิวในคำสั่งนี้ได้มีดังนี้ เส้นตรง จุด เส้นโค้ง วงกลม วงรี เส้นโพลีไลน์ และ เส้นสไปน์ รูปที่ 2.21 แสดงตัวอย่างการสร้างรูปผิวด้วยคำสั่ง RuleSurf



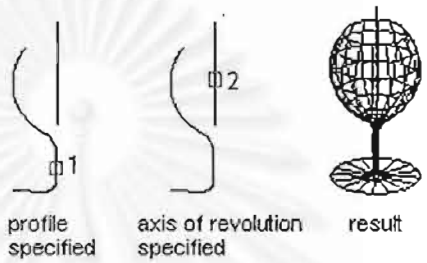
รูปที่ 2.21 ภาพแสดงการสร้างรูปผิวจากคำสั่ง RuleSurf

4. คำสั่ง TabSurf เป็นการสร้างรูปผิว Tabulated ซึ่งสร้างจากเส้น Path คือ เส้นที่กำหนดลักษณะรูปร่างของรูปผิว และเส้นทิศทาง (Direction Curve) ซึ่งเป็นเส้นที่กำหนดทิศทางและขนาดความยาวของรูปผิว โดยที่เส้นทิศทางสามารถเป็นได้เฉพาะเส้นตรงกับเส้นโพลีไลน์เท่านั้น และขั้นตอนการสร้างรูปผิว Tabulated แสดงไว้ที่รูปที่ 2.22



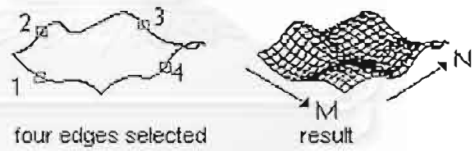
รูปที่ 2.22 ภาพแสดงการสร้างรูปผิวจากคำสั่ง TabSurf

5. คำสั่ง RevSurf เป็นการสร้างรูปผิว Revolute โดยสร้างจากหมุนเส้นพาท (Path) รอบแกนที่กำหนด เส้นพาทจะเป็นตัวกำหนดทิศทางของเส้น N ซึ่งสามารถจะเป็นได้ทั้งเส้นตรง เส้นโค้ง วงกลม วงรี เส้นโพลีไลน์ และ เส้นสไปน์ ถ้าเส้น Path เป็นเส้นปิด เช่น วงกลม วงรี เป็นต้น ก็จะทำให้รูปผิวที่ได้เป็นโพลีกอนเมชที่มี N ปิด และแกนที่เป็นจุดหมุนจะเป็นได้เฉพาะเส้นตรงกับโพลีไลน์เท่านั้น รูปที่ 2.23 แสดงตัวอย่างการสร้างรูปผิว Revolute ด้วยคำสั่ง RevSurf



รูปที่ 2.23 ภาพแสดงการสร้างรูปผิวจากคำสั่ง RevSurf

6. คำสั่ง EdgeSurf เป็นการสร้าง Coons Surface Patch โดยสร้างจากเส้น 4 เส้นที่ประกอบกันเป็นรูปทรงปิด เส้นแต่ละเส้นจะถูกเรียกว่า Edge และรูปที่ 2.24 แสดงตัวอย่างรูปผิว EdgeSurf



รูปที่ 2.24 ภาพแสดงการสร้างรูปผิวจากคำสั่ง EdgeSurf

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 3
โครงสร้างและขั้นตอนการใช้งานโปรแกรม

โปรแกรมที่ผู้วิจัยได้พัฒนาเป็นโปรแกรมช่วยในการผลิต หรือ CAM (Computer Aided Manufacturing) ซึ่งใช้สำหรับงานกัด (Milling) ในลักษณะ 3 แกนที่เป็นรูปผิว (Surface) ด้วยเครื่องกัดซีเอ็นซี (CNC Milling Machine) ของ FANUC Model MA15 นอกจากนี้ผู้วิจัยได้ทำการพัฒนาโปรแกรมใช้สำหรับงานกัดในลักษณะ 2 แกนครึ่งด้วย เพื่อที่จะได้ชุดโปรแกรมช่วยในการผลิตที่สมบูรณ์ยิ่งขึ้น โดยโปรแกรมที่ผู้วิจัยได้พัฒนาแบ่งออกเป็น 5 โปรแกรม ดังนี้

1. โปรแกรมสำหรับงานเจาะ (Drilling) โดยจะรับแบบชิ้นงานที่เป็นจุด (Point) และวงกลม (Circle) เท่านั้น ในการแปลงแบบชิ้นงานให้เป็นรหัสจี (G-Code) จะใช้จุดที่ได้จากรูปวาดจุด หรือจุดศูนย์กลาง (Center Point) ของรูปวาดวงกลม มาเป็นจุดศูนย์กลางของรูที่จะทำการเจาะ

2. โปรแกรมสำหรับงานกัดหยาบ (Rough Milling) แบบ 2 แกนครึ่ง จะเป็นการกัดเอาเนื้อวัสดุที่ไม่ต้องการออก ให้เหลือแต่บริเวณที่ต้องการ แต่ยังไม่เป็นรูปลักษณะที่ต้องการ และการกัดจะเป็นแบบ 2 แกนครึ่ง คือ ขณะกัดวัสดุหัวกัด (End Mill) จะเคลื่อนที่เฉพาะแกน X และ Y เท่านั้น แกน Z จะถูกกำหนดให้คงที่ หัวกัดที่ใช้กัดในโปรแกรมนี้ถูกกำหนดให้ใช้กับหัวแบบ Flat คำสั่งนี้จะรับแบบชิ้นงานเส้นโค้งปิด (Closed Curve) ได้ทุกชนิด เช่น วงกลม วงรี (Ellipse) และโพลีไลน์ (Polyline) เป็นต้น

3. โปรแกรมสำหรับงานกัดละเอียด (Contour Milling) แบบ 2 แกนครึ่ง จะเป็นการกัดเพื่อเก็บรายละเอียดของชิ้นงานให้ได้ชิ้นงานตามที่ต้องการ หลังจากกัดวัสดุด้วยโปรแกรมสำหรับงานกัดหยาบ โปรแกรมนี้จะรับแบบชิ้นงานที่เป็น เส้นตรง (Line) ส่วนโค้ง (Arc) วงกลม วงรี และโพลีไลน์

4. โปรแกรมสำหรับงานกัดหยาบแบบ 3 แกน ซึ่งจะกัดวัสดุเฉพาะด้านบนที่ไม่ต้องการของวัสดุชิ้นงานให้เหลือเฉพาะบริเวณที่จะกัดละเอียด โดยรับแบบชิ้นงานผิว เว้นแต่ว่าแบบชิ้นงานผิวใดที่เป็นรูปทรงปิด เช่น รูปผิวทรงกลม (Sphere) รูปผิวกล่อง (Box) เป็นต้น ก็จะไม่สามารถแปลงเป็นรหัสจีได้

5. โปรแกรมสำหรับงานกัดละเอียดแบบ 3 แกน คือขณะกัดวัสดุหัวกัดสามารถเคลื่อนที่ได้ทั้ง 3 แกนพร้อมๆ กัน และการใช้คำสั่งนี้จะใช้กัดชิ้นวัสดุต่อจากโปรแกรมสำหรับกัด

หยาบของการกัดแบบ 3 แกนเพื่อให้ได้รูปทรงของชิ้นงานที่ต้องการ ในคำสั่งนี้จะใช้หัวกัดแบบหัวบอล (Ball) และรับแบบชิ้นงานกับโปรแกรมสำหรับกัดหยาบของการกัดแบบ 3 แกน

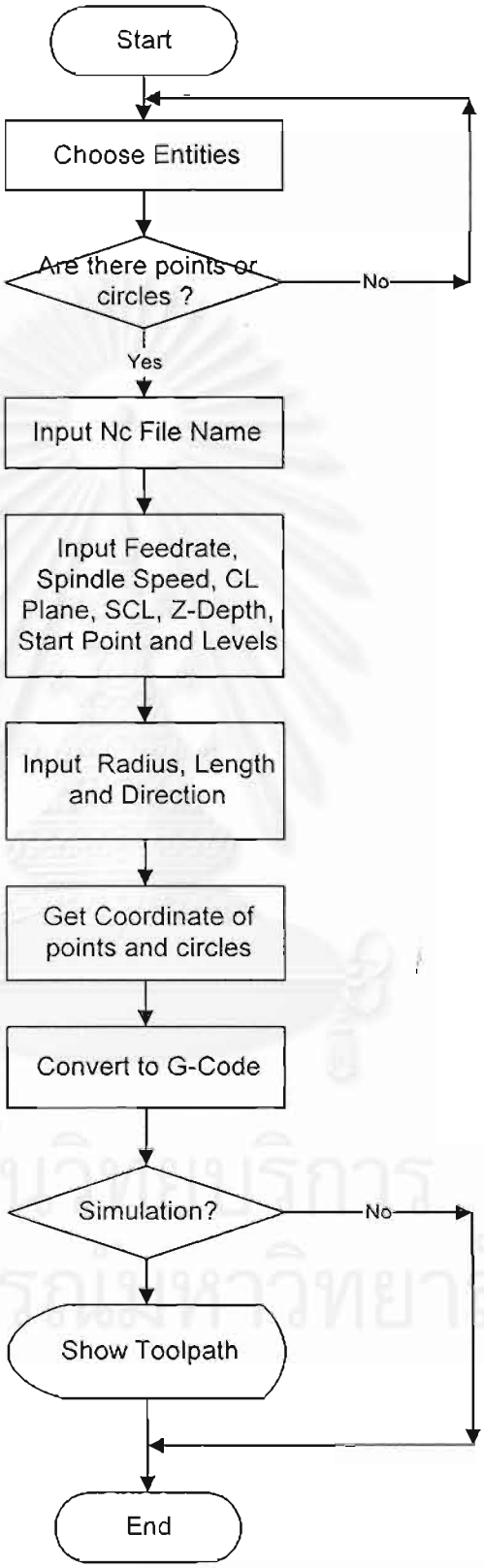
โดยที่ทั้ง 5 โปรแกรมนี้จะทำงานร่วมกับโปรแกรมออโตแคดรีลีส 14 (AutoCAD Release 14) ซึ่งจะเป็นฟังก์ชันของโปรแกรมออโตแคด การเขียนโปรแกรมทั้ง 5 โปรแกรมจะใช้โปรแกรม Visual C++ Version 6.0 เป็นตัวคอมไพเลอร์ (Compiler) และใช้ตัวต่อประสาน (Interface) ARX (AutoCAD Runtime Extension) ช่วยในการเข้าถึงระบบและข้อมูลภายในโปรแกรมออโตแคด

3.1 โครงสร้างและการออกแบบโปรแกรม

3.1.1 โปรแกรมสำหรับงานเจาะ

ผู้วิจัยได้ออกแบบโปรแกรมให้สามารถรับแบบชิ้นงานที่วาดจากโปรแกรมออโตแคดที่เป็นจุดและวงกลมเท่านั้นและสามารถเลือกแบบชิ้นงานได้ครั้งละหลายรูปได้ตามที่ผู้ใช้ต้องการ ผู้ใช้สามารถเลือกรูปวาดอื่นที่ไม่ใช่จุดและวงกลมได้ โปรแกรมจะทำการคัดเลือกเฉพาะชนิดรูปวาดที่โปรแกรมต้องการเท่านั้น โปรแกรมจะรับค่าพิกัด X และ Y จากจุดและจุดศูนย์กลางของวงกลมมาเป็นพิกัดที่ใช้ในการเขียนรหัสจี และภายในโปรแกรมนี้ยังมีพารามิเตอร์ให้เลือกว่ารูที่จะเจาะนั้นต้องการเจาะรูละกี่ครั้ง โดยแต่ละครั้งของการเจาะจะมีค่าความลึกที่ต่างกัน เพื่อต้องการคายเศษวัสดุจากการเจาะและป้องการขูดขีดผิวของรูจากเศษวัสดุจากการเจาะเพื่อให้ได้คุณภาพของผิวรูที่ได้มีลักษณะผิวที่ดี และพารามิเตอร์ของหัวกัดไม่ใช้ในการเขียนรหัสจีแต่มีไว้ใช้ในการจำลองเส้นทางเดินของหัวเจาะเท่านั้น เพื่อจะได้จำลองขนาดของหัวเจาะได้ถูกต้องตามความต้องการของผู้ใช้ ซึ่งโครงสร้างขั้นตอนการทำงานของโปรแกรมสำหรับงานเจาะแสดงไว้ด้วยรูปที่ 3.1

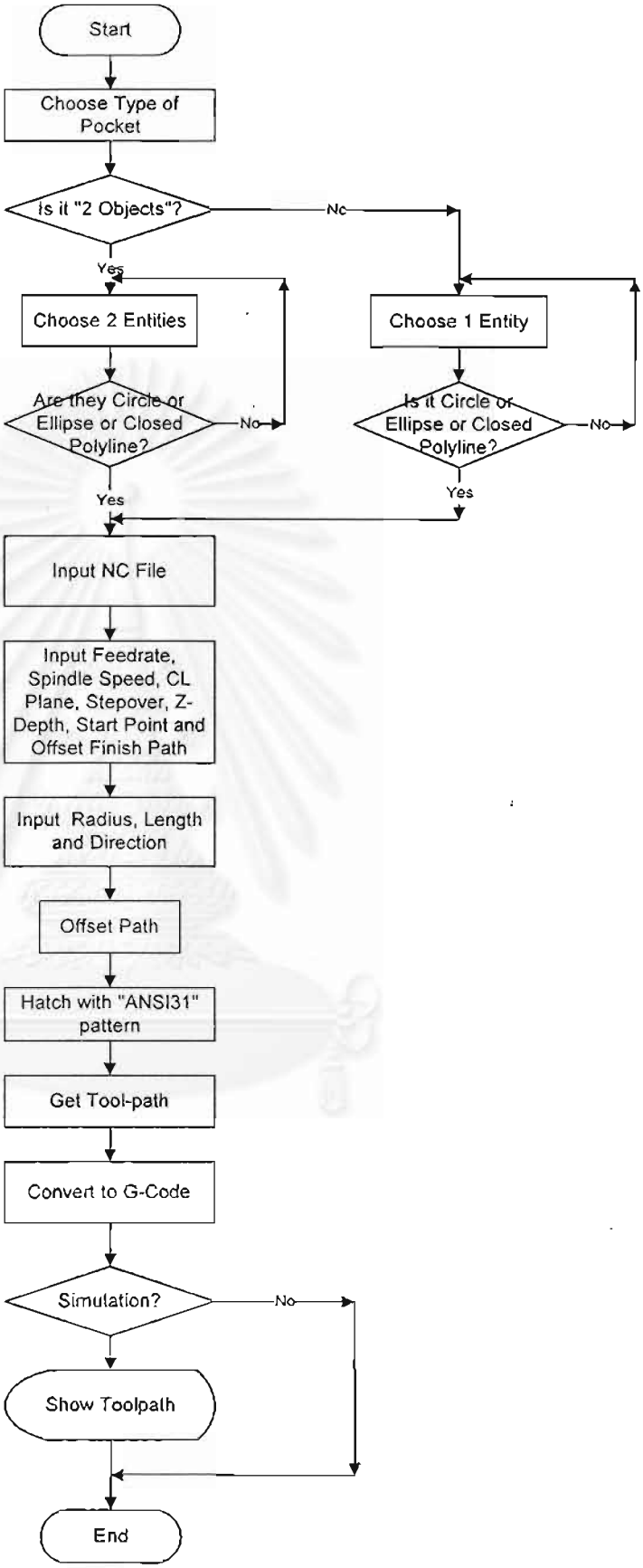
สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย



รูปที่ 3.1 ภาพแสดงแผนผังของโปรแกรมสำหรับงานเจาะ

3.1.2 โปรแกรมสำหรับงานกัดหยาบแบบ 2 แกนครึ่ง

ผู้วิจัยได้ออกแบบโปรแกรมให้สามารถรับแบบชิ้นงานที่วาดจากโปรแกรมออโตแคดที่เป็นรูปวาดปิด ได้แก่ วงกลม วงรี และ โพลีไลน์ที่ปิด เท่านั้น โดยมีรูปแบบของการกัดอยู่ 2 แบบด้วยกันคือ รูปแบบที่ 1) "2 Object" จะเป็นการกัดเอาเนื้อวัสดุระหว่างเส้นโค้งปิด 2 เส้น โดยเส้นโค้งปิดแรกจะเป็นเส้นแบบชิ้นงานที่ต้องการกัดซึ่งเป็นเส้นแบบชิ้นงานเป้าหมายของการกัดและเส้นโค้งปิดแรกนี้จะต้องอยู่ภายในบริเวณของเส้นโค้งปิดที่สอง ส่วนเส้นโค้งปิดที่สองจะเป็นบริเวณขอบเขตของวัสดุที่จะกัดหรือวัดถุดิบก่อนกัด รูปแบบที่ 2) "1 Object" จะเป็นการกัดเนื้อวัสดุบริเวณภายในของแบบชิ้นงานที่ต้องการกัด การหาเส้นทางเดินของหัวกัด (Tool path) ของทั้ง 2 รูปแบบจะใช้คำสั่ง "Hatch" ผ่านฟังก์ชัน (Function) ของ ARX โดยใช้รูปแบบของการ Hatch เป็น ANSI31 ซึ่งเป็นรูปแบบของการ Hatch มาตรฐานที่มีอยู่ในโปรแกรมออโตแคด และมีมุมเอียงของเส้น Hatch เท่ากับ 45 องศา ก่อนที่จะทำการ Hatch ก็จะออฟเซต (Offset) แบบชิ้นงานที่ต้องการกัดก่อนให้เท่าค่าของรัศมีของหัวกัดรวมกับค่าออฟเซตเผื่อออกห่างจากแบบชิ้นงานที่ต้องการกัด ซึ่งทิศทางการออฟเซตของรูปแบบการกัดแรกจะออฟเซตเส้นแบบชิ้นงานที่ต้องการให้โตออก ซึ่งจะได้เส้นออฟเซตที่มีขนาดใหญ่กว่าแบบชิ้นงานที่ต้องการ และส่วนการออฟเซตของรูปแบบการกัดแบบที่ 2 จะออฟเซตเส้นแบบชิ้นงานที่ต้องการให้มีขนาดเล็กลง ซึ่งการออฟเซตจะเป็นการป้องกันไม่หัวกัดเข้ากัดบริเวณวัสดุที่ต้องการ และการ Hatch รูปแบบการกัดแบบแรกจะ Hatch ระหว่างแบบชิ้นงานที่ได้จากการออฟเซตกับแบบชิ้นงานขอบเขตของวัสดุ ส่วนรูปแบบที่ 2 จะ Hatch ภายในแบบชิ้นงานที่ได้จากการ ออฟเซตเมื่อทำการ Hatch เสร็จก็จะได้จุดจากการ Hatch จากนั้นก็นำจุดเหล่านั้นนำมาจัดเรียงเป็นเส้นทางการเดินของหัวกัด จากนั้นก็นำเส้นทางเดินของหัวกัดที่ได้นำไปเขียนรหัสจี ผู้วิจัยได้เขียนแผนผังการทำงานไว้ดังแสดงที่รูปที่ 3.2

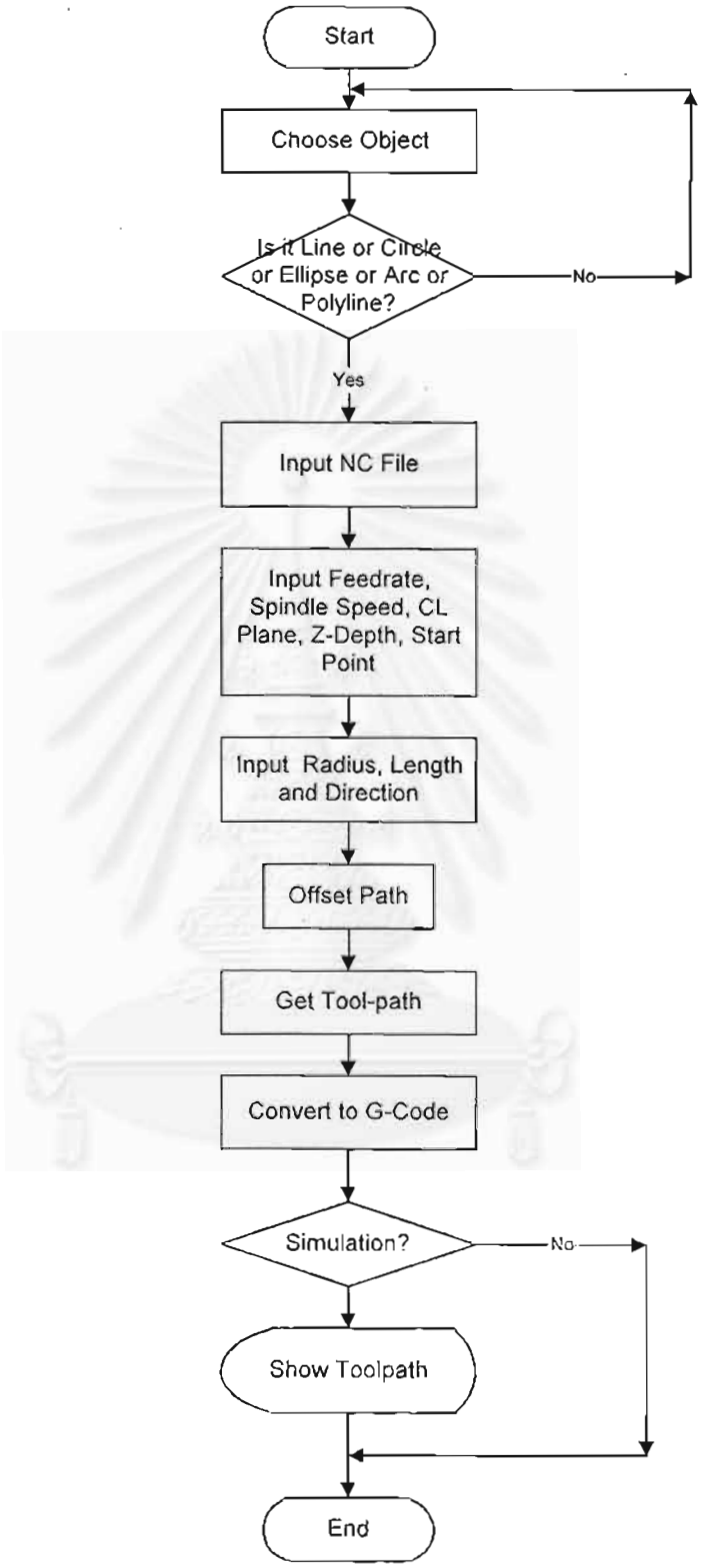


รูปที่ 3.2 ภาพแสดงแผนผังของโปรแกรมสำหรับงานกัดหยาบแบบ 2 แกนครึ่ง

3.1.3 โปรแกรมสำหรับงานกัดละเอียดแบบ 2 แกนครึ่ง

ผู้วิจัยได้ออกแบบให้โปรแกรมรับแบบชิ้นงานที่เป็นเส้นตรง เส้นโค้ง วงกลม วงรี และ โพลีไลน์ การหาเส้นทางการเดินของหัวกัดขั้นแรกก็จะทำการออฟเซตแบบชิ้นงานที่ต้องการ โดยระยะออฟเซตจะเท่ากับค่ารัศมีของหัวกัด ซึ่งการออฟเซตของรูปเส้นตรงจะได้รูปเส้นตรง รูปเส้นโค้งจะได้เส้นโค้ง รูปวงกลมก็จะได้วงกลม รูปโพลีไลน์ก็จะได้โพลีไลน์ส่วนวงรีจะได้สไปน์ (Spline) จากนั้นก็ทำการอ่านและรับข้อมูลภายในของแบบชิ้นงานที่ได้จากการออฟเซตเช่น เส้นตรงก็จะรับค่าจุดเริ่มต้นและจุดสุดท้าย เป็นต้น แต่สำหรับแบบชิ้นงานออฟเซตที่เป็นสไปน์จะแตกต่างจากรูปชนิดอื่นๆ คือ รูปที่เป็นสไปน์จะต้องคำนวณหาค่าจุดที่อยู่บนเส้นสไปน์ โดยจะต้องกำหนดระยะทางระหว่างจุดที่ต้องการมาคำนวณหาจุดบนเส้นสไปน์ และเมื่อโปรแกรมได้รับข้อมูลของแบบชิ้นงานที่ได้จากการออฟเซตแล้วก็จะตรวจสอบว่าแบบชิ้นงานที่ได้จากออฟเซตเป็นอะไร เช่นถ้าเป็นเส้นตรงก็จะเขียนรหัสจีด้วย G01 แล้วตามด้วยตำแหน่ง ถ้าเป็นวงกลมหรือเส้นโค้งก็จะเขียนรหัสจีด้วย G02 ส่วนวงรีก็จะเขียนด้วย G01 จากนั้นก็นำรหัสจีที่ได้ไปเขียนต่อกันเป็นไฟล์รหัสจี ซึ่งขั้นตอนการทำงานของโปรแกรมได้แสดงไว้ในรูปที่ 3.3

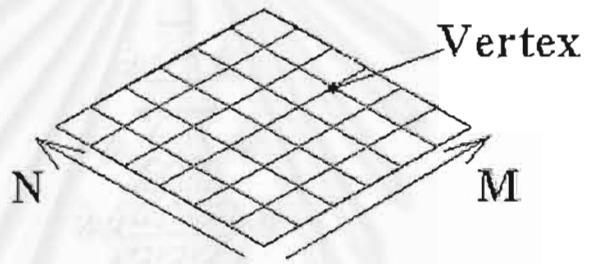
สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย



รูปที่ 3.3 ภาพแสดงแผนผังของโปรแกรมสำหรับงานกัดละเอียดแบบ 2 แกนครึ่ง

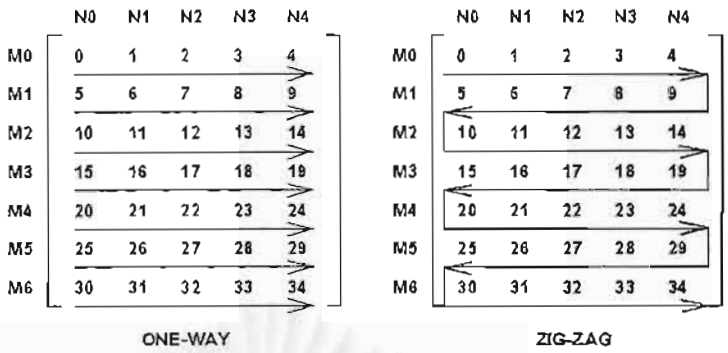
3.1.4 โปรแกรมสำหรับงานกัดหยาบและงานกัดละเอียดแบบ 3 แกน

ผู้วิจัยได้ออกแบบโปรแกรมทั้ง 2 โปรแกรมให้รับเอนทิตีจากโปรแกรมออโตแคดเฉพาะที่เป็นรูปผิว โดยจะรับรูปผิวได้ครั้งละ 1 รูปเท่านั้น แต่ทว่ารูปผิวที่วาดจากโปรแกรมออโตแคดจะเก็บข้อมูลเป็นโพลีกอนเมช (Polygonmesh) ซึ่งจะเก็บเฉพาะจุดบนโพลีกอนเมชและประกอบไปด้วยเส้น M และ N เรียงไขว่กันทำให้เกิดเป็นตารางสี่เหลี่ยมเรียกว่า ตาข่าย (Mesh) และจุดที่ได้จากการตัดกันของเส้นทั้งสองเรียกว่า จุดยอด (Vertex) ตามรูปที่ 3.4 ดังนั้นจึงจำเป็นที่จะต้องสร้างสมการของผิวจากจุดยอดที่ได้จากโพลีกอนเมช ซึ่งผู้วิจัยได้เลือกที่จะสร้างสมการผิวด้วย NURBS (Non Uniform Rational B - Spline) เนื่องด้วยโปรแกรม CAD/CAM ส่วนใหญ่ได้ใช้รูปผิว NURBS เป็นตัวสร้างงานผิว และ ARX ก็มีฟังก์ชันในการสร้างรูปผิว NURBS ด้วย



รูปที่ 3.4 ภาพแสดงโพลีกอนเมชจากการวาดรูปผิวด้วยโปรแกรมออโตแคด

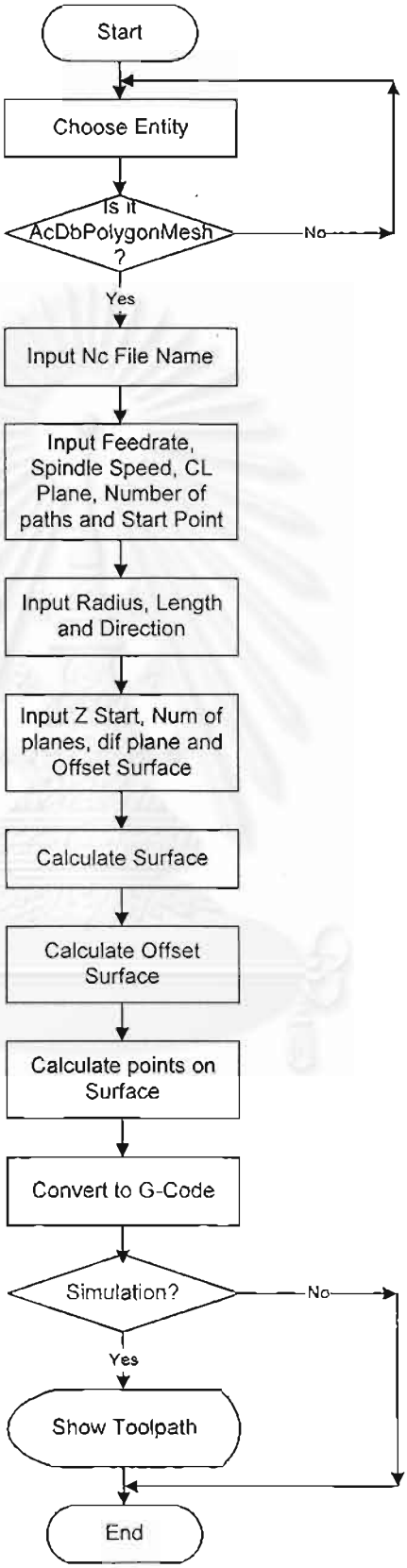
หลังจากสร้างรูปผิว NURBS แล้วก็ต้องนำเอารูปผิว NURBS นี้ ไปทำการออฟเซตให้เท่ากับค่ารัศมีของหัวกัด ก็เพราะถ้าไม่ออฟเซตผิวงาน เมื่อกัดชิ้นงานจริงแล้วจะได้ชิ้นงานที่เล็กกว่าขนาดที่วาดขึ้นเท่ากับรัศมีของหัวกัด เมื่อได้รูปผิวออฟเซต NURBS (Offset NURBS Surface) แล้วก็นำรูปผิวออฟเซต NURBS มาคำนวณหาจุดรูปผิวออฟเซต NURBS ซึ่งทั้งการออฟเซตและการคำนวณหาจุดบนผิว ผู้วิจัยใช้ฟังก์ชันของ ARX มาช่วยในการคำนวณเช่นกัน เมื่อได้จุดบนผิวงานต่อไปก็นำจุดเหล่านั้นไปจัดเรียงเพื่อให้ได้ทิศทางการเดินของหัวกัด (Tool path) ตามที่ต้องการ ซึ่งทั้ง 2 โปรแกรมที่พัฒนาขึ้นจะมีโครงสร้างการทำงานที่คล้ายกัน แต่ต่างกันที่การจัดเรียงจุดบนผิวงาน โดยที่โปรแกรมสำหรับกัดละเอียดจะเรียงจุดเป็นแบบเมทริกซ์ (Matrix) และจำแนกออกเป็น 2 ชนิด คือ 1) แบบทิศทางเดียว (One-Way) 2) แบบไปกลับ (Zig-Zag) โดยแสดงการจัดเรียงจุดทั้ง 2 แบบไว้ที่รูปที่ 3.5



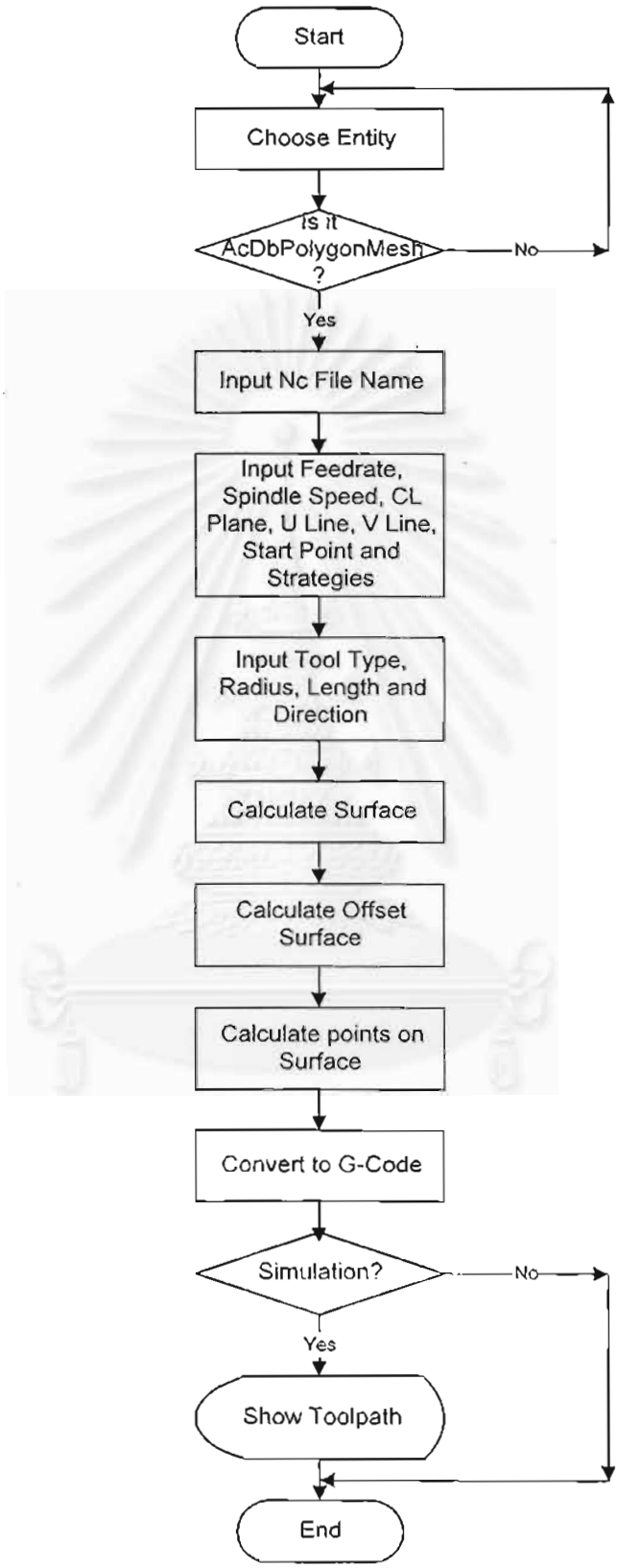
รูปที่ 3.5 ภาพแสดงการจัดเรียงจุดของการกัดละเอียด

แต่สำหรับโปรแกรมสำหรับกัดหยาบจะเป็นการกัดทีละระนาบ (Plane) โดยกัดที่ค่า Z คงที่ เมื่อกัดที่ค่า Z ที่สูงกว่าเสร็จก็จะกัดที่ระนาบที่มีค่า Z ต่ำกว่า และจะกัดจากระนาบ Z มากไปหาระนาบ Z น้อยเสมอ โดยการเรียงจุดนั้นจะเป็นการเรียงแบบไปกลับ และแต่ละระนาบ Z ก็จะนำจุดบนรูปผิวออฟเซต NURBS มาเปลี่ยนค่า Z ให้เป็นค่า Z ของระนาบ Z นั้นๆ ก็จะได้จุดบนแต่ละระนาบ จากนั้นก็จะนำจุดแต่ละระนาบมาเปรียบเทียบกับจุดบนรูปผิวออฟเซต NURBSโดยเปรียบเทียบกันเฉพาะค่า Z เท่านั้น และการเปรียบเทียบของจุด 2 จุดจะต้องมีค่า X และ Y ที่ตรงกัน การเปรียบเทียบจะเปรียบเทียบว่าถ้าค่า Z ของระนาบมากกว่าค่า Z ของรูปผิวออฟเซต NURBS ก็จะนำค่าจุดบนระนาบไปเขียนรหัสจี แต่ถ้าน้อยกว่าก็จะข้ามจุดนั้นไปและไม่เขียนรหัสจีของจุดนั้น และในส่วนท้ายของโปรแกรมทั้ง 2 จะเป็นการแสดงเส้นทางการเดินของหัวกัด ในส่วนนี้โปรแกรมจะอ่านรหัสจีจากไฟล์รหัสจีที่โปรแกรมได้เขียนขึ้นโดยตรง เพื่อเป็นการตรวจสอบการเขียนรหัสจีที่เขียนจากโปรแกรมในส่วนต้นว่าถูกต้องหรือไม่ จากที่กล่าวมาทั้งหมดนี้ก็เป็นแนวทางการเขียนโปรแกรมที่ผู้วิจัยได้พัฒนาขึ้น ผู้วิจัยได้เขียนแผนผังของการพัฒนาโปรแกรมไว้ตามรูปที่ 3.6 และ 3.7

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย



รูปที่ 3.6 ภาพแสดงแผนผังของโปรแกรมสำหรับงานกัดหยาบแบบ 3 แกน

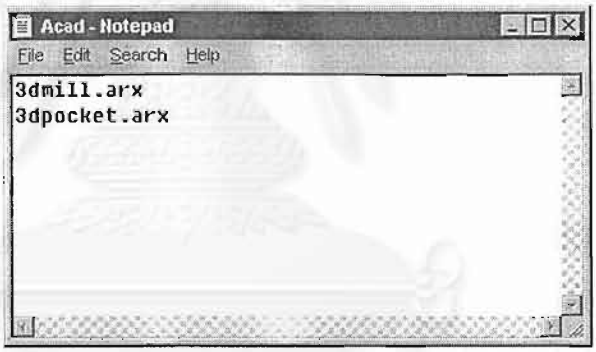


รูปที่ 3.7 ภาพแสดงแผนผังของโปรแกรมสำหรับงานกัดละเอียดแบบ 3 แกน

3.2 การติดตั้งและการโหลด (Load) โปรแกรม ARX

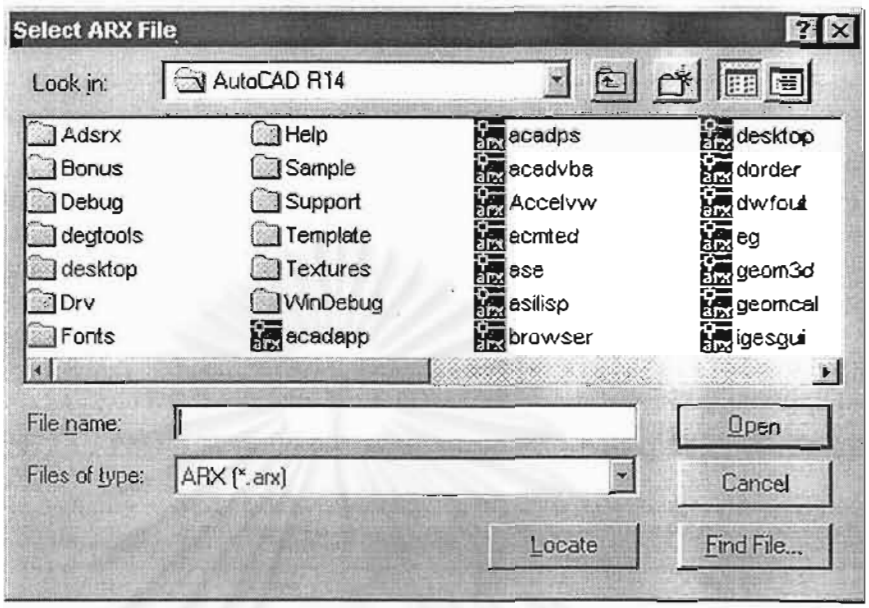
ก่อนอื่นให้ทำการคัดลอกไฟล์ (Copy) หรือเคลื่อนที่ไฟล์ (Move) ARX ที่ต้องการไปไว้อยู่ในไดเรกทอรี่ (Directory) หรือแฟ้ม (Folder) ของโปรแกรมออโตแคด โดยในวิทยานิพนธ์ฉบับนี้ได้ตั้งชื่อไฟล์ ARX สำหรับกัดหยาบว่า 3dpocket.arx และสำหรับกัดละเอียดว่า 3dmill.arx โดยที่การโหลดไฟล์ ARX กระทำได้ 2 วิธี คือ

1. ให้โปรแกรมออโตแคดโหลดให้เองโดยอัตโนมัติเมื่อเปิดโปรแกรมออโตแคด วิธีนี้จะต้องสร้างไฟล์ชื่อ Acad.rx ขึ้นมาก่อน ซึ่งเป็น text ไฟล์ และกำหนดให้ไฟล์ Acad.rx อยู่ภายในแฟ้มของโปรแกรมออโตแคด จากนั้นให้เปิดไฟล์ Acad.rx ด้วยโปรแกรม NotePad หรือ Editor ตัวใดก็ได้แล้วให้พิมพ์ชื่อไฟล์ ARX พร้อมนามสกุล ARX ที่ต้องการให้โปรแกรมออโตแคดโหลดให้อย่างอัตโนมัติลงไปในไฟล์ Acad.rx โดยพิมพ์ชื่อไฟล์ ARX ชื่อละบรรทัดตามรูปที่ 3.8 จากนั้นก็บันทึกไฟล์ ต่อไปเมื่อเปิดโปรแกรมออโตแคด โปรแกรมออโตแคดก็จะทำการอ่านข้อมูลภายในไฟล์ Acad.rx ก่อน แล้วก็จะโหลดไฟล์ ARX ให้อย่างอัตโนมัติ



รูปที่ 3.8 ภาพแสดงการพิมพ์ชื่อไฟล์ ARX ด้วยโปรแกรมโน้ตแพด

2. โหลดไฟล์ ARX ด้วยตนเอง ให้พิมพ์คำว่า "ARX" ที่ Command Line ในโปรแกรมออโตแคด จากนั้นให้พิมพ์ "L" แล้วกด Enter ก็จะมีไดอะล็อกบ็อกซ์ (Dialog Box) ขึ้นมาตามรูปที่ 3.9 แล้วพิมพ์ชื่อไฟล์ ARX หรือเลือกไฟล์ ARX ที่ปรากฏอยู่ภายในไดอะล็อกบ็อกซ์ จากนั้นกดปุ่ม Open โปรแกรมออโตแคดก็จะทำการโหลดไฟล์ ARX ให้



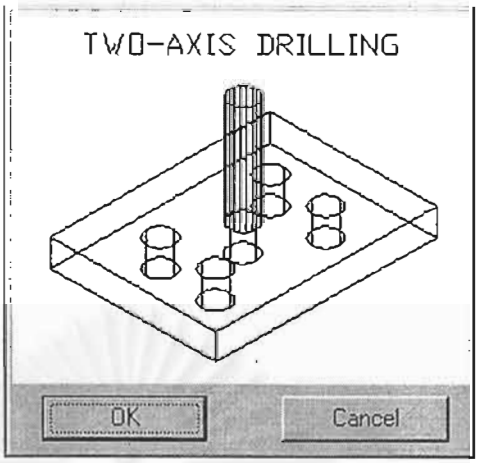
รูปที่ 3.9 ภาพแสดงไดอะล็อกบ็อกซ์ของการโหลดไฟล์ ARX ด้วยตนเอง

3.3 การใช้งานโปรแกรมที่พัฒนา

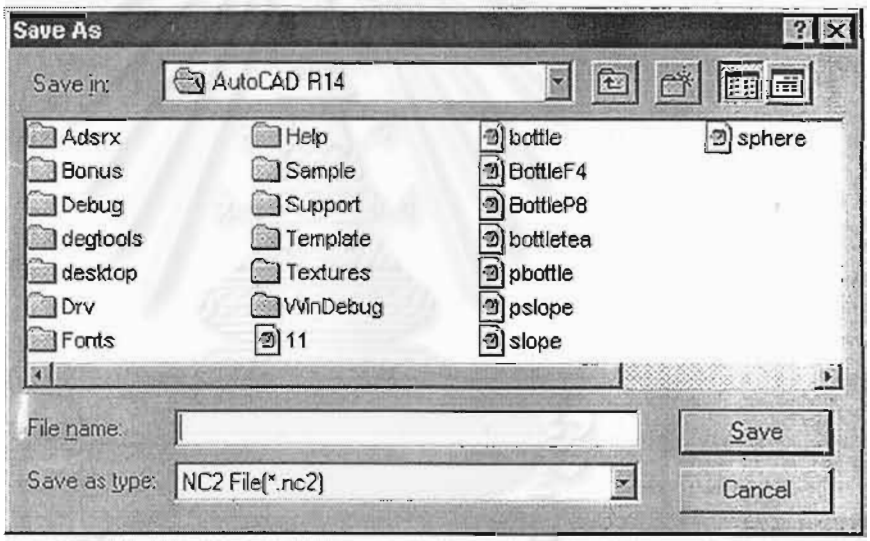
3.3.1 โปรแกรมสำหรับงานเจาะ

ผู้วิจัยได้ตั้งชื่อฟังก์ชันนี้ว่า 2dDrill เพื่อเรียกใช้งานโปรแกรมสำหรับงานเจาะ โดยที่ไฟล์ ARX ก็ตั้งชื่อเดียวกันกับชื่อฟังก์ชัน ขั้นแรกพิมพ์ 2dDrill ที่ Command Line จากนั้นก็จะมีไดอะล็อกบ็อกซ์ปรากฏขึ้นดังรูปที่ 3.10 ถ้าต้องการใช้งานโปรแกรมนี้ก็ให้กดปุ่ม OK แต่ถ้าไม่ต้องการใช้งานก็ให้กดปุ่ม Cancel ก็จะเป็นการจบการทำงานของโปรแกรมนี้ และเมื่อกดปุ่ม OK แล้วโปรแกรมก็จะให้เลือกรูปวาดตามที่ผู้ใช้ต้องการโดยจะแสดงข้อความที่ Command Line ว่า "Select objects :" เมื่อเลือกชิ้นงานเสร็จเรียบร้อยแล้วโปรแกรมจะถามผู้ใช้เพื่อให้แน่ใจว่าใช่ชิ้นงานที่ต้องการเจาะ โดยแสดงข้อความที่ Command Line ว่า "Do you want this object -> <Yes>/No:" ถ้าผู้ใช้เลือกถูกต้องแล้วให้กดคีย์ Enter เพื่อทำงานต่อไป แต่ทว่าไม่ใช่ชิ้นงานที่ต้องการก็ให้กดคีย์ N ตามด้วย Enter โปรแกรมก็จะให้เลือกชิ้นงานใหม่อีกครั้ง เมื่อผ่านขั้นตอนนี้แล้วโปรแกรมจะแสดงไดอะล็อกบ็อกซ์ ซึ่งเป็นไดอะล็อกบ็อกซ์ให้ใส่ชื่อไฟล์และเลือกแฟ้มที่ต้องการตามรูปที่ 3.11 โดยชื่อไฟล์ที่ใส่ไปจะเป็นไฟล์ของรหัสจีที่ได้จากการใช้งานจากโปรแกรมนี้ เมื่อใส่ชื่อไฟล์ตามที่ต้องการเสร็จแล้ว โปรแกรมก็จะให้ผู้ใช้ใส่ค่าพารามิเตอร์ (Parameters) ต่างๆ ผ่านไดอะล็อกบ็อกซ์ในรูปที่ 3.12 และ 3.13 ดังต่อไปนี้

I19397563



รูปที่ 3.10 ภาพแสดงไดอะล็อกบ็อกซ์แรกก่อนเข้าใช้โปรแกรมสำหรับงานเจาะ



รูปที่ 3.11 ภาพแสดงไดอะล็อกบ็อกซ์ให้เติมชื่อไฟล์รหัสจี

1. Start Point เป็นจุดเริ่มต้นของการเจาะชิ้นงาน ซึ่งจะต้องเติมค่าทั้ง 3 แกน คือ X, Y และ Z โดยมีค่าปริยาย (Default) ที่ X=0, Y=0 และ Z=100

2. Feed Rate เป็นอัตราการป้อน คือความเร็วในการเคลื่อนที่ของดอกสว่านขณะเจาะชิ้นงาน มีหน่วยเป็น มิลลิเมตรต่อนาที (mm/min)

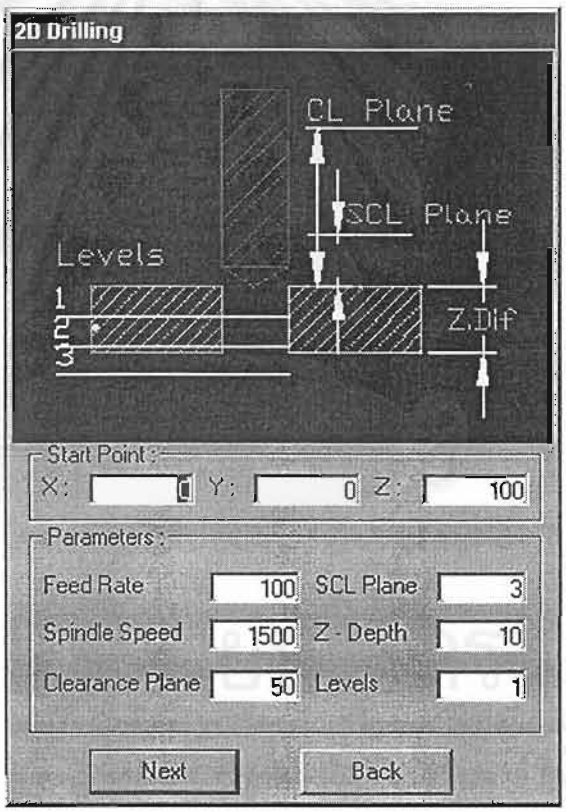
3. Spindle Speed เป็นอัตราการหมุนของดอกสว่านขณะกัดชิ้นงาน มีหน่วยเป็น รอบต่อนาที (rev/min)

4. Clearance Plane เป็นระนาบที่มีค่าของแกน Z คงที่ โดยที่ระนาบนี้ดอกสว่านสามารถเคลื่อนที่ด้วยความเร็วสูงสุดของเครื่องโดยปลอดภัยจากการที่ดอกสว่านวิ่งเข้าชนกับชิ้นงาน มีหน่วยเป็น มิลลิเมตร (mm)

5. SCL Plane ย่อมาจาก Sub Clearance Plane เป็นระนาบของแกน Z ที่ปลอดภัยขณะที่ดอกสว่านเคลื่อนที่เร็วจากจุดที่เจาะอยู่ไปยังจุดที่จะเจาะต่อไป มีหน่วยเป็นมิลลิเมตร (mm)

6. Z-Depth เป็นระยะความลึกที่ต้องการจะเจาะวัสดุ มีหน่วยเป็น มิลลิเมตร (mm)

7. Levels เป็นจำนวนชั้นของการเจาะ

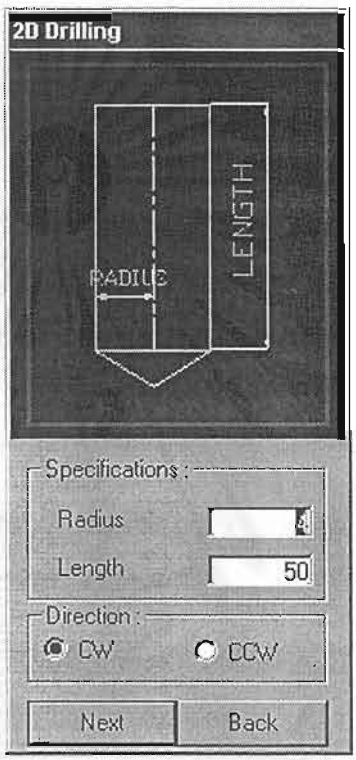


รูปที่ 3.12 ภาพแสดงไดอะล็อกบ็อกซ์ให้ใส่ข้อมูลของโปรแกรมสำหรับงานเจาะ

8. Radius เป็นค่ารัศมีของดอกสว่าน มีหน่วยเป็น มิลลิเมตร (mm)

9. Length เป็นค่าความยาวของดอกสว่าน มีหน่วยเป็น มิลลิเมตร (mm)

10. Direction เป็นการเลือกทิศทางการหมุนของดอกสว่าน มีให้เลือกอยู่ 2 แบบ คือ หมุนตามเข็มนาฬิกา (CW) และหมุนทวนเข็มนาฬิกา (CCW)



รูปที่ 3.13 ภาพแสดงไดอะล็อกบ็อกซ์เพื่อเติมข้อมูลเกี่ยวกับดอกสว่าน

เมื่อใส่ข้อมูลต่างๆ เรียบร้อย โปรแกรมก็จะทำการหาเส้นทางการเดินของดอกสว่านให้อย่างอัตโนมัติ จากนั้นก็จะถามผู้ใช้ว่าต้องการให้แสดงการเคลื่อนที่ของดอกสว่านหรือไม่ ซึ่งจะแสดงข้อความที่ Command Line ว่า "Do you want to simulate: Yes/<No>" ถ้าต้องการให้แสดงการเคลื่อนที่ของดอกสว่านก็กดคีย์ "Y" ตามด้วย Enter โปรแกรมก็จะแสดงการเคลื่อนที่ของดอกสว่าน แต่ถ้าไม่ต้องการแสดงก็ให้กด Enter เมื่อผ่านขั้นตอนนี้แล้วก็จะจบการทำงานของโปรแกรมสำหรับงานเจาะ ผู้ใช้ก็จะได้รหัสจีอยู่ในแฟ้มที่ผู้ใช้ได้กำหนดไว้

3.3.2 โปรแกรมสำหรับงานกัดหยาบแบบ 2 แกนครึ่ง

ชื่อฟังก์ชันสำหรับใช้เรียกโปรแกรมสำหรับงานกัดหยาบแบบ 2 แกนครึ่งผู้วิจัยได้ตั้งชื่อว่า 2dPocket และตั้งชื่อไฟล์ ARX ว่า 2dPocket.arx ในการเรียกใช้ให้พิมพ์คำว่า "2DPocket" ที่ Command Line จากนั้นก็จะมีไดอะล็อกบ็อกซ์ปรากฏขึ้นดังรูปที่ 3.14 ถ้าต้องการใช้งานโปรแกรมนี้ก็ให้กดปุ่ม OK แต่ถ้าไม่ต้องการใช้งานก็ให้กดปุ่ม Cancel ก็จะเป็นการจบการทำงานของโปรแกรมนี้ และเมื่อกดปุ่ม OK แล้วโปรแกรมก็จะแสดงไดอะล็อกบ็อกซ์ตามรูปที่ 3.15 เพื่อที่จะให้ผู้ใช้เลือกรูปแบบการกัด โดยมีรูปแบบการกัดให้เลือก 2 รูปแบบดังได้กล่าวไว้แล้วในหัวข้อที่ 3.1.2 จากนั้นโปรแกรมก็จะให้เลือกรูปวาดที่ต้องการ ถ้าผู้ใช้เลือกรูปแบบการกัดเป็นแบบ 2 Objects ผู้ใช้จะต้องเลือกรูปวาดที่ต้องการ 2 รูป โดยรูปแรกจะเป็นรูปที่ผู้ต้องการกัดวัสดุให้เป็นตามรูปที่เลือก ส่วนรูปที่ 2 จะเป็นรูปวาดที่ใช้แสดงขอบเขตของเนื้อวัสดุที่จะทำการกัด แต่ถ้าผู้ใช้เลือกรูปแบบการกัดเป็นแบบ 1 Object ผู้ใช้จะต้องเลือกรูปวาดที่ต้องการ 1 รูป จากนั้นโปรแกรมก็จะให้ใส่ชื่อไฟล์รหัสจีลงในไดอะล็อกบ็อกซ์ตามรูปที่ 3.11 เมื่อใส่ชื่อไฟล์ตามที่ต้องการเสร็จแล้ว โปรแกรมก็จะให้ผู้ใช้ใส่ค่าพารามิเตอร์ต่างๆ ผ่านไดอะล็อกบ็อกซ์ในรูปที่ 3.16 และ 3.17 ดังต่อไปนี้

1. Start Point เป็นจุดเริ่มต้นของการเจาะชิ้นงาน ซึ่งจะต้องเติมค่าทั้ง 3 แกน คือ X, Y และ Z โดยมีค่าปริยายที่ X=0, Y=0 และ Z=100

2. Feed Rate เป็นอัตราการป้อน คือความเร็วในการเคลื่อนที่ของหัวกัดขณะเจาะชิ้นงาน มีหน่วย.ป็น มิลลิเมตรต่อนาที (mm/min)

3. Spindle Speed เป็นอัตราการหมุนของหัวกัดขณะกัดชิ้นงาน มีหน่วยเป็นรอบต่อนาที (rev/min)

4. Clearance Plane เป็นระนาบที่มีค่าของแกน Z คงที่ โดยที่ระนาบนี้หัวกัดสามารถเคลื่อนที่ด้วยความเร็วสูงสุดของเครื่องโดยปลอดภัยจากการที่ดอกสว่านวิ่งเข้าชนกับชิ้นงาน มีหน่วยเป็น มิลลิเมตร (mm)

5. Stepover เป็นระยะทางระหว่างเส้นทางเดินของหัวกัด โดยวัดจากจุดศูนย์กลางของหัวกัดกับจุดศูนย์กลางของหัวกัด มีหน่วยเป็น มิลลิเมตร (mm)

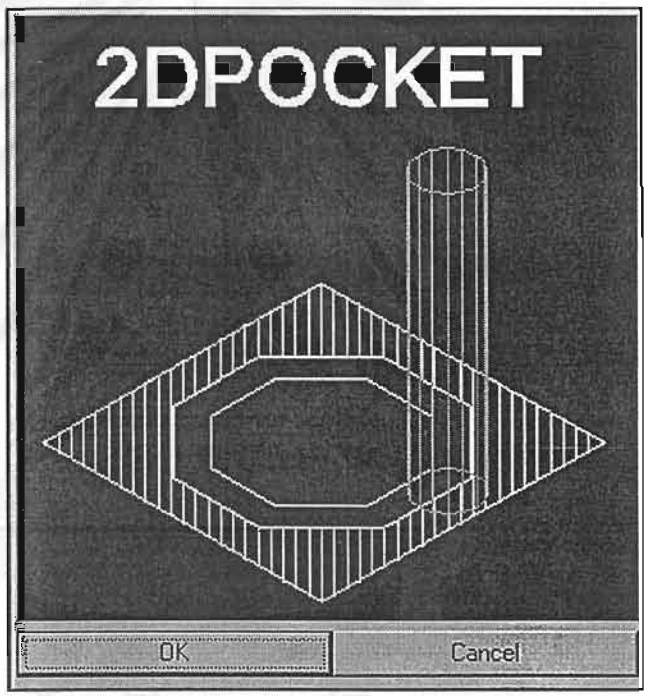
6. Z-Depth เป็นระยะความลึกที่ต้องการจะกัดวัสดุ มีหน่วยเป็น มิลลิเมตร (mm)

7. Offset Finish Path เป็นค่าระยะเผื่อออกจากรูปวาดที่ต้องการ มีหน่วยเป็น มิลลิเมตร (mm)

8. Radius เป็นค่ารัศมีของหัวกัด มีหน่วยเป็น มิลลิเมตร (mm)

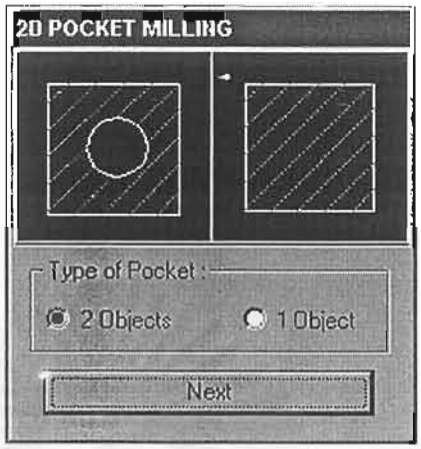
9. Length เป็นค่าความยาวของหัวกัด มีหน่วยเป็น มิลลิเมตร (mm)

10. Direction เป็นการเลือกทิศทางการหมุนของหัวกัด มีให้เลือกอยู่ 2 แบบ คือ หมุนตามเข็มนาฬิกา (CW) และหมุนทวนเข็มนาฬิกา (CCW)

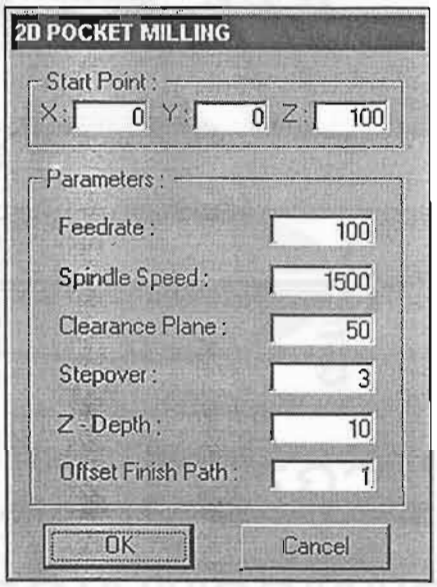


รูปที่ 3.14 ภาพแสดงไดอะล็อกบ็อกซ์แรกก่อนเข้าใช้โปรแกรมสำหรับงานกัดหยาบแบบ 2 แกนครึ่ง

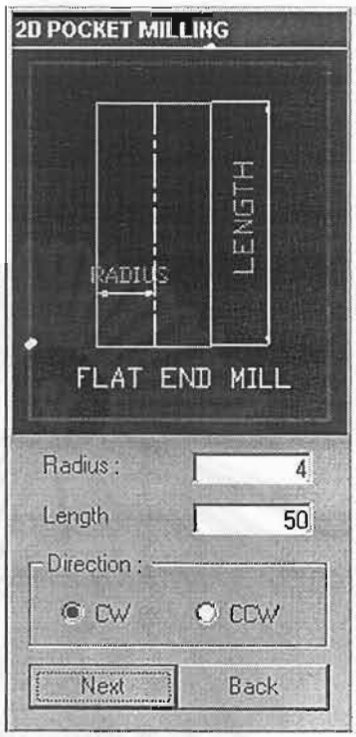
สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย



รูปที่ 3.15 ภาพแสดงไดอะล็อกบ็อกซ์สำหรับเลือกรูปแบบการกัดของโปรแกรมสำหรับงานกัดหยาบแบบ 2 แกนครึ่ง



รูปที่ 3.16 ภาพแสดงไดอะล็อกบ็อกซ์ให้ใส่ข้อมูลของโปรแกรมสำหรับงานกัดหยาบแบบ 2 แกนครึ่ง



รูปที่ 3.17 ภาพแสดงไดอะล็อกบ็อกซ์เพื่อเติมข้อมูลเกี่ยวกับหัวกัด

3.1.3 โปรแกรมสำหรับงานกัดละเอียดแบบ 2 แกนครึ่ง

ผู้วิจัยได้กำหนดชื่อไฟล์ ARX กับชื่อฟังก์ชันไว้เหมือนกันคือ 2dContour เริ่มแรกให้พิมพ์ชื่อฟังก์ชันที่ Command Line จากนั้นโปรแกรมก็จะแสดงไดอะล็อกบ็อกซ์เกิดขึ้นตามรูปที่ 3.18 ถ้าต้องการใช้งานโปรแกรมนี้ก็ให้กดปุ่ม OK แต่ถ้าไม่ต้องการใช้งานก็ให้กดปุ่ม Cancel ก็จะเป็นการจบการทำงานของโปรแกรมนี้ และเมื่อกดปุ่ม OK แล้วโปรแกรมก็จะให้เลือกรูปที่ต้องการ จากนั้นโปรแกรมก็จะให้เลือกด้านของรูปที่ต้องการจะกัด จากนั้นก็จะให้ใส่ชื่อไฟล์รหัสจีดามรูปที่ 3.11 ซึ่งต่อไปโปรแกรมก็จะถามผู้ใช้ให้ใส่ค่าตัวแปรต่างๆ ด้วยไดอะล็อกบ็อกซ์ในรูปที่ 3.19 และ 3.20 ดังต่อไปนี้

1. Start Point เป็นจุดเริ่มต้นของการเจาะชิ้นงาน ซึ่งจะต้องเติมค่าทั้ง 3 แกน คือ X, Y และ Z โดยมีค่าปริยายที่ X=0, Y=0 และ Z=100

2. Feed Rate เป็นอัตราการป้อน คือความเร็วในการเคลื่อนที่ของดอกสว่านขณะเจาะชิ้นงาน มีหน่วยเป็น มิลลิเมตรต่อนาที (mm/min)

3. Spindle Speed เป็นอัตราการหมุนของดอกสว่านขณะกัดชิ้นงาน มีหน่วยเป็น รอบต่อนาที (rev/min)

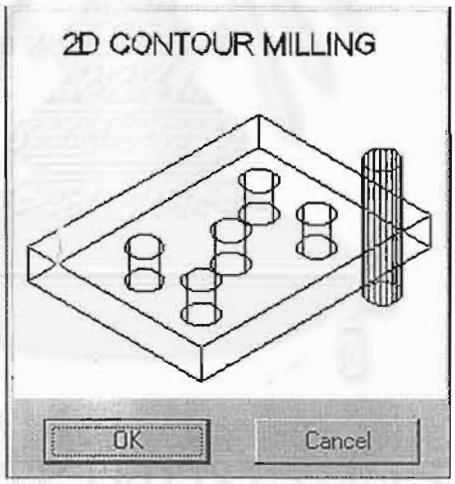
4. Clearance Plane เป็นระนาบที่มีค่าของแกน Z คงที่ โดยที่ระนาบนี้หัวกัดสามารถเคลื่อนที่ด้วยความเร็วสูงสุดของเครื่องโดยปลอดภัยจากการที่หัวกัดวิ่งเข้าชนกับชิ้นงาน มีหน่วยเป็น มิลลิเมตร (mm)

5. Z-Depth เป็นระยะความลึกที่ต้องการจะกัด มีหน่วยเป็น มิลลิเมตร (mm)

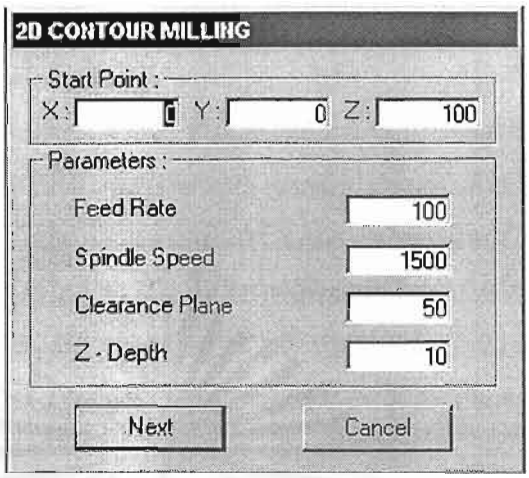
6. Radius เป็นค่ารัศมีของหัวกัด มีหน่วยเป็น มิลลิเมตร (mm)

7. Length เป็นค่าความยาวของหัวกัด มีหน่วยเป็น มิลลิเมตร (mm)

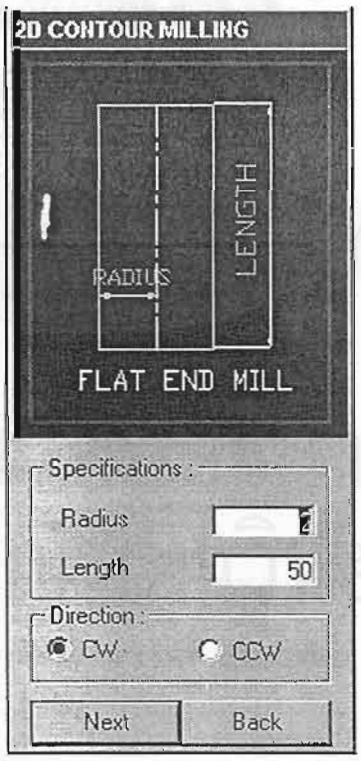
8. Direction เป็นการเลือกทิศทางการหมุนของหัวกัด มีให้เลือกอยู่ 2 แบบ คือ หมุนตามเข็มนาฬิกา (CW) และหมุนทวนเข็มนาฬิกา (CCW)



รูปที่ 3.18 ภาพแสดงไดอะล็อกบ็อกซ์แรกก่อนเข้าใช้โปรแกรมสำหรับงานกัดละเอียดแบบ 2 แกนครึ่ง



รูปที่ 3.19 ภาพแสดงไดอะล็อกบ็อกซ์ให้ใส่ข้อมูลของโปรแกรมสำหรับงานกัดละเอียดแบบ 2 แกนครึ่ง



รูปที่ 3.20 ภาพแสดงไดอะล็อกบ็อกซ์เพื่อเติมข้อมูลเกี่ยวกับหัวกัด

3.3.4 โปรแกรมสำหรับงานกัดหยาบของการกัดแบบ 3 แกน

ผู้วิจัยได้ตั้งชื่อฟังก์ชันสำหรับเรียกใช้โปรแกรมสำหรับกัดหยาบนี้ว่า 3dPocket และชื่อไฟล์ ARX ไว้ว่า 3dPocket.arx ขั้นแรกให้พิมพ์ "3dPocket" ที่ Command Line จากนั้นจากมีไดอะล็อกบ็อกซ์แสดงขึ้นมาดังรูปที่ 3.21 จากนั้นโปรแกรมก็จะให้เลือกผิวชิ้นงาน เมื่อผ่านขั้นตอนนี้แล้วโปรแกรมจะแสดงไดอะล็อกบ็อกซ์ ซึ่งเป็นไดอะล็อกบ็อกซ์ให้ใส่ชื่อไฟล์รหัสจีและเลือกแฟ้มที่ต้องการตามรูปที่ 3.11 เมื่อใส่ชื่อไฟล์ตามที่ต้องการเสร็จแล้ว โปรแกรมก็จะให้ผู้ใช้ใส่ค่าพารามิเตอร์ต่างๆ ผ่านไดอะล็อกบ็อกซ์ตามรูปที่ 3.22 ถึง 3.24 ดังต่อไปนี้

1. Start Point เป็นจุดเริ่มต้นของการวิ่งเข้าหาชิ้นงาน

2. Feed Rate เป็นอัตราการป้อน หรืออัตราเร็วของการเคลื่อนที่ของหัวกัดขณะกัดชิ้นงาน มีหน่วยเป็น มิลลิเมตรต่อนาที (mm/min)

3. Spindle Speed เป็นอัตราการหมุนของหัวกัดขณะกัดชิ้นงาน มีหน่วยเป็นรอบต่อนาที (rev/min)

4. Clearance Plane เป็นระนาบที่มีค่าของแกน Z คงที่ โดยที่ระนาบนี้หัวกัดสามารถเคลื่อนที่ด้วยความเร็วสูงสุดของเครื่องโดยปลอดภัยจากการที่หัวกัดวิ่งเข้าชนกับชิ้นงาน มีหน่วยเป็น มิลลิเมตร (mm)

5. Number of Paths เป็นจำนวนเส้นทางเดินของหัวกัดในแต่ละระนาบ

6. Radius เป็นค่ารัศมีของหัวกัด ซึ่งในโปรแกรมนี้จะใช้เฉพาะหัวแบบแฟลดเท่านั้น มีหน่วยเป็น มิลลิเมตร (mm)

7. Length เป็นค่าความยาวของหัวกัด มีหน่วยเป็น มิลลิเมตร (mm)

8. Direction เป็นการเลือกทิศทางการหมุนของหัวกัด ซึ่งมีให้เลือกอยู่ 2 แบบ คือ แบบตามเข็มนาฬิกา (CW) และแบบทวนเข็มนาฬิกา (CCW)

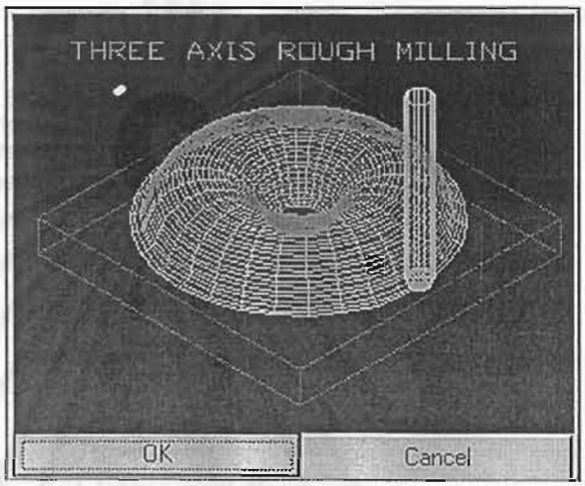
9. Z. Start Plane เป็นค่าระนาบของ Z ที่เริ่มต้นของการกัด

10. Num. of Plane เป็นจำนวนของระนาบที่ต้องการกัดหยาบ

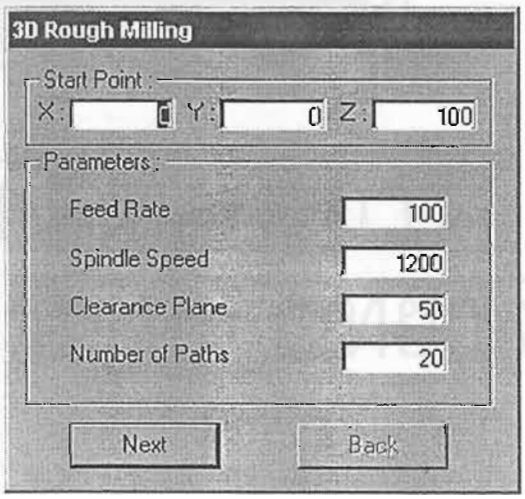
11. Dif. Plane เป็นค่าระยะห่างระหว่างระนาบกับระนาบที่ใช้ในการกัด มีหน่วยเป็น มิลลิเมตร (mm)

12. Offset Surface เป็นค่าระยะที่ให้หัวกัดกัดห่างจากผิวงานที่ต้องการ มีหน่วยเป็น มิลลิเมตร (mm)

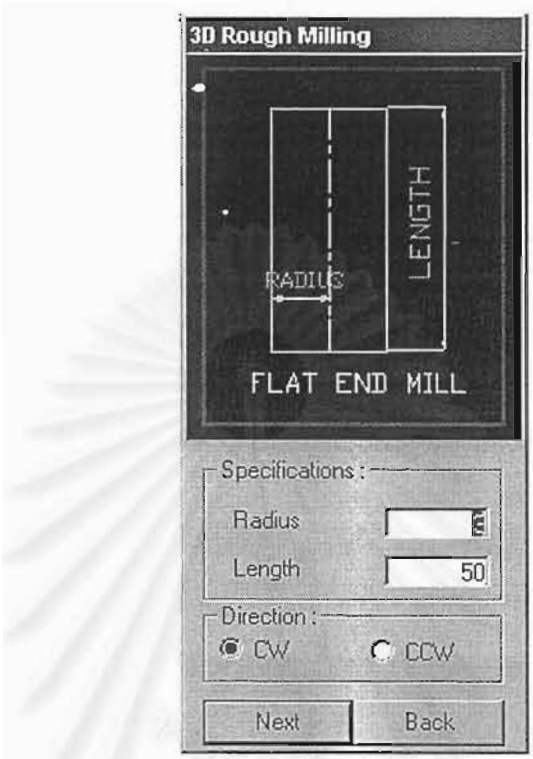
ซึ่งเมื่อเติมพารามิเตอร์เสร็จแล้วโปรแกรมก็จะทำการคำนวณหาเส้นทางเดินของหัวกัด เมื่อโปรแกรมคำนวณเสร็จก็จะถามผู้ใช้ว่าจะแสดงการจำลองเส้นทางเดินของหัวกัดหรือไม่ เมื่อจำลองการเดินของหัวกัดเสร็จก็จะสิ้นสุดการใช้โปรแกรมนี้



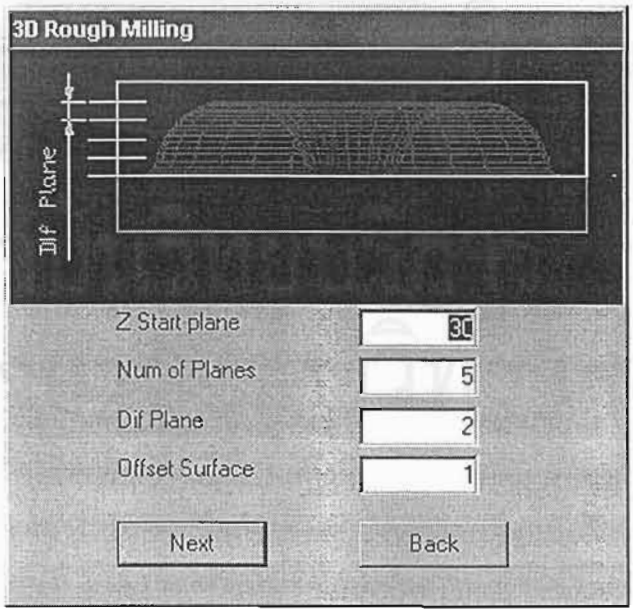
รูปที่ 3.21 ภาพแสดงไดอะล็อกบ็อกซ์แรกก่อนเข้าใช้โปรแกรมสำหรับงานกัดหยาบแบบ 3 แกน



รูปที่ 3.22 ภาพแสดงไดอะล็อกบ็อกซ์เพื่อใส่ข้อมูลของโปรแกรมสำหรับงานกัดหยาบแบบ 3 แกน



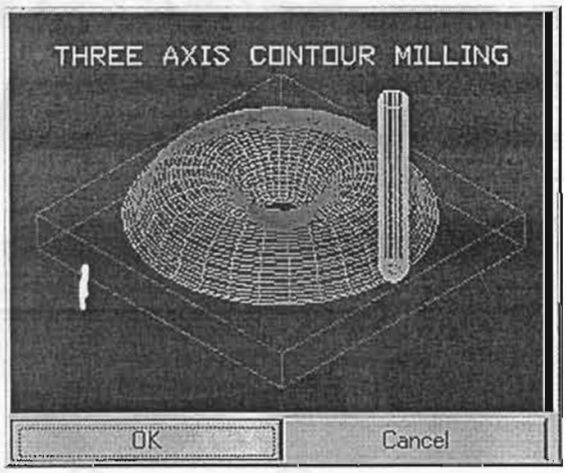
รูปที่ 3.23 ภาพแสดงไดอะล็อกบ็อกซ์เพื่อเติมข้อมูลเกี่ยวกับหัวกัด



รูปที่ 3.24 ภาพแสดงไดอะล็อกบ็อกซ์เพื่อเติมข้อมูลเกี่ยวกับระนาบ

3.3.5 โปรแกรมสำหรับงานกัดละเอียดของการกัดแบบ 3 แกน

ผู้วิจัยได้ตั้งชื่อฟังก์ชันไว้สำหรับเรียกใช้โปรแกรมนี้ในโปรแกรมออโตแคดว่า 3dContour ดังนั้นเมื่อเปิดโปรแกรมออโตแคดแล้วก็ทำการโหลดไฟล์ 3dContour.arx ตามวิธีที่ได้กล่าวมาแล้ว จากนั้นพิมพ์ 3dContour ลงใน Command Line เพื่อเรียกใช้โปรแกรมสำหรับกัดละเอียด เมื่อกดคีย์ Enter แล้วจะมีไดอะล็อกบ็อกซ์เกิดขึ้นตามรูปที่ 3.25 ถ้าต้องการใช้งานโปรแกรมนี้ก็ให้กดปุ่ม OK แต่ถ้าไม่ต้องการใช้งานก็ให้กดปุ่ม Cancel ก็จะเป็นการจบการทำงานของโปรแกรมนี้ และเมื่อกดปุ่ม OK แล้วโปรแกรมก็จะให้เลือกผิวชิ้นงานที่ต้องการจะกัดโดยจะแสดงข้อความที่ Command Line ว่า "Select an entity:" การเลือกชิ้นงานจะต้องเลือกชิ้นงานที่สร้างจากคำสั่ง Surface เท่านั้น และรูปงานผิวที่วาดจะต้องไม่เป็นรูปทรงปิด เช่นรูปทรงกลม เป็นต้น แต่ถ้าเลือกชิ้นงานที่ไม่ใช่รูปงานผิว โปรแกรมก็จะให้เลือกชิ้นงานใหม่ เมื่อเลือกชิ้นงานเสร็จเรียบร้อยแล้วโปรแกรมก็จะเปลี่ยนสีของชิ้นงานนั้นเพื่อแสดงให้ทราบว่าเลือกชิ้นงานนี้ และจะถามผู้ใช้เพื่อให้แน่ใจว่าใช่ชิ้นงานที่ต้องการกัด โดยแสดงข้อความที่ Command Line ว่า "Do you want this object -> <Yes>/No:"



รูปที่ 3.25 ภาพแสดงไดอะล็อกบ็อกซ์แรกก่อนเข้าใช้โปรแกรมสำหรับงานกัดละเอียดแบบ 3 แกน

ถ้าผู้ใช้เลือกถูกต้องแล้วให้กดคีย์ Enter เพื่อทำงานต่อไป แต่ทว่าไม่ใช่ชิ้นงานที่ต้องการก็ให้กดคีย์ N ตามด้วย Enter โปรแกรมก็จะให้เลือกชิ้นงานใหม่อีกครั้ง เมื่อผ่านขั้นตอนนี้แล้วโปรแกรมจะแสดงไดอะล็อกบ็อกซ์ ซึ่งเป็นไดอะล็อกบ็อกซ์ให้ใส่ชื่อไฟล์รหัสจีและเลือกแฟ้มที่ต้องการตามรูปที่ 3.11 เมื่อใส่ชื่อไฟล์ตามที่ต้องการเสร็จแล้ว โปรแกรมก็จะให้ผู้ใช้ใส่ค่าพารามิเตอร์ต่างๆ ผ่านไดอะล็อกบ็อกซ์ตามรูปที่ 3.26 ถึง 3.28 ดังต่อไปนี้

1. Start Point เป็นจุดเริ่มต้นของการกัดชิ้นงาน ซึ่งจะต้องเติมค่าทั้ง 3 แกน คือ X, Y และ Z โดยมีค่าปริยายที่ X=0, Y=0 และ Z=100

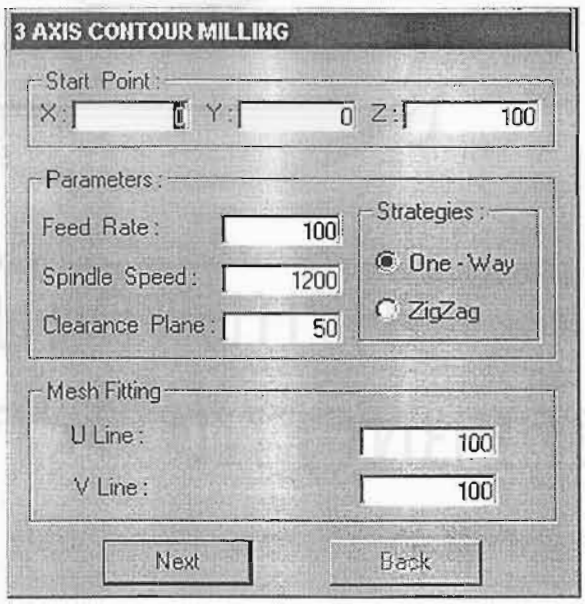
2. Feed Rate เป็นอัตราการป้อน คือความเร็วในการเคลื่อนที่ของหัวกัดขณะกัดชิ้นงาน มีหน่วยเป็น มิลลิเมตรต่อนาที (mm/min)

3. Spindle Speed เป็นอัตราการหมุนของหัวกัดขณะกัดชิ้นงาน มีหน่วยเป็นรอบต่อนาที (rev/min)

4. Clearance Plane เป็นระนาบที่มีค่าของแกน Z คงที่ โดยที่ระนาบนี้หัวกัดสามารถเคลื่อนที่ด้วยความเร็วสูงสุดของเครื่องโดยปลอดภัยจากการที่หัวกัดวิ่งเข้าชนกับชิ้นงาน มีหน่วยเป็น มิลลิเมตร (mm)

5. Mesh Fitting เป็นการกำหนดค่าความละเอียดในการกัดชิ้นงาน โดยระบุเป็นจำนวนเส้นในแนวแกน U และแนวแกน V โดยเติมค่าภายในช่องของ U Line และ V Line ตามลำดับ

6. Strategies เป็นการเลือกลักษณะการเคลื่อนที่ของหัวกัดขณะกัดชิ้นงาน ซึ่งมีให้เลือกอยู่ 2 แบบคือ 1. แบบทิศทางเดียว 2. แบบไปกลับ



รูปที่ 3.26 ภาพแสดงไดอะล็อกบ็อกซ์ให้ใส่ข้อมูลของโปรแกรมสำหรับงานกัดละเอียดแบบ 3 แกน

7. Tool Type เป็นการเลือกชนิดของหัวกัด ซึ่งจะมีอยู่ 2 แบบ คือ แบบหัวบอล (Ball Mill) กับ แบบหัวแฟลต (Flat Mill) โดยที่โปรแกรมนี้จะยังใช้ได้เฉพาะแบบหัวบอลได้เท่านั้น

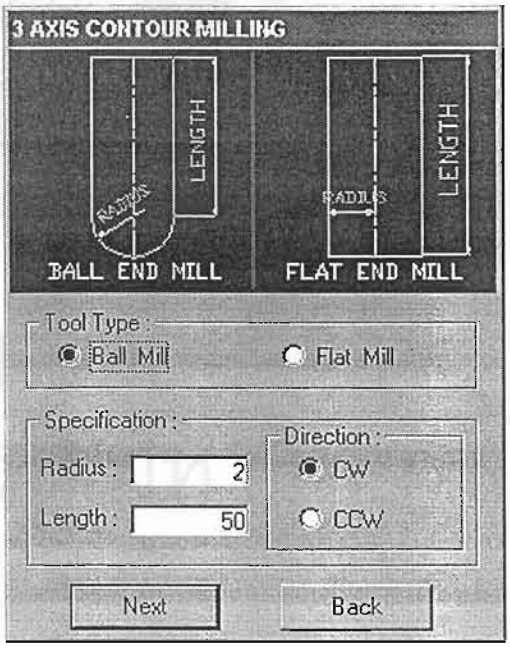
8. Radius เป็นค่ารัศมีของหัวกัด มีหน่วยเป็น มิลลิเมตร (mm)

9. Length เป็นค่าความยาวของหัวกัด มีหน่วยเป็น มิลลิเมตร (mm)

10. Direction เป็นการเลือกทิศทางการหมุนของหัวกัด มีให้เลือกอยู่ 2 แบบ คือ หมุนตามเข็มนาฬิกา (CW) และหมุนทวนเข็มนาฬิกา (CCW)

11. Approach เป็นการเลือกลักษณะการเข้ากัดชิ้นงานของหัวกัด ซึ่งมีให้เลือกอยู่ 3 แบบ คือ 1. Along Z Axis เป็นการเข้ากัดชิ้นงานตามแกน Z 2. Normal Surf. เป็นการเข้ากัดชิ้นงานโดยตั้งฉากกับผิวชิ้นงาน 3. Arc to Surf. เป็นการเข้ากัดชิ้นงานในลักษณะเส้นโค้งสัมผัสกับชิ้นงาน ซึ่งการเข้ากัดชิ้นงานแบบนี้ยังไม่สามารถใช้งานได้

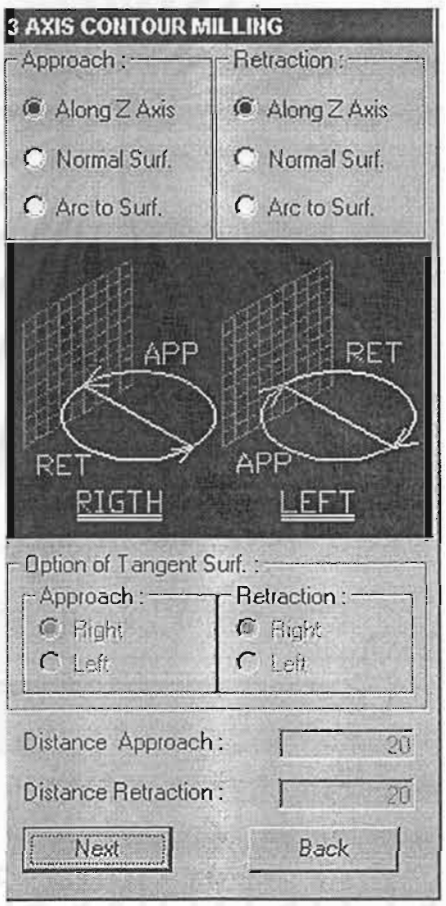
12. Retraction เป็นการเลือกลักษณะการออกจากชิ้นงานของหัวกัด ซึ่งมีวิธีการออกเหมือนกับการเข้ากัดชิ้นงาน



รูปที่ 3.27 ภาพแสดงไดอะล็อกบ็อกซ์เพื่อให้ใส่ข้อมูลเกี่ยวกับหัวกัด

13. Distance Approach เป็นระยะทางระหว่างผิวชิ้นงานกับตำแหน่งของหัวกัดที่จะเริ่มเข้ากัดชิ้นงาน โดยจะเติมค่านี้ก็ต่อเมื่อเลือกลักษณะการเข้ากัดชิ้นงานเป็นแบบ Normal Surf. กับ Arc to Surf. เท่านั้น

14. Distance Retraction เป็นระยะทางระหว่างผิวชิ้นงานกับตำแหน่งของหัวกัดที่ออกจากชิ้นงาน โดยจะเติมค่านี้ก็ต่อเมื่อเลือกลักษณะการเข้ากัดชิ้นงานเป็นแบบ Normal Surf. กับ Arc to Surf. เท่านั้น



รูปที่ 3.28 ภาพแสดงไดอะล็อกบ็อกซ์เพื่อเลือกรูปแบบการเข้าและออกชิ้นงาน

เมื่อใส่ข้อมูลต่างๆครบตามที่โปรแกรมถาม โปรแกรมก็จะคำนวณหาเส้นทางเดินของหัวกัดตามที่ผู้ใช้ได้เลือกไว้ จากนั้นก็จะแสดงเส้นทางเดินของหัวกัดออกมา พร้อมกับถามผู้ใช้ว่าต้องการให้แสดงการเคลื่อนที่ของหัวกัดหรือไม่ ซึ่งจะแสดงข้อความที่ Command Line ว่า "Do you want to simulate: Yes/<No>" ถ้าต้องการให้แสดงการเคลื่อนที่ของหัวกัดก็กดคีย์ "Y" ตามด้วย Enter โปรแกรมก็จะแสดงการเคลื่อนที่ของหัวกัด แต่ถ้าไม่ต้องการแสดงก็

ให้กด Enter เมื่อผ่านขั้นตอนนี้แล้วก็จะจบการทำงานของโปรแกรมสำหรับกัดละเอียด ผู้ใช้ก็จะได้รหัสจีอยู่ในแฟ้มที่ผู้ใช้ได้กำหนดไว้

บทที่ 4
การทดสอบโปรแกรมและผลการทดสอบ

การทดสอบโปรแกรมที่พัฒนาผู้วิจัยได้ลำดับขั้นตอนในการทดสอบ 3 ขั้นตอน คือ

1. ขั้นตอนการใช้โปรแกรมในการหาเส้นทางเดินของหัวกัด (Tool path)

ในขั้นตอนนี้ผู้วิจัยได้กำหนดแบบชิ้นงานที่ต้องการทดสอบไว้ทั้งงานกัดแบบ 2 แกนครึ่ง และ 3 แกน รวมถึงงานเจาะด้วย โดยบางแบบชิ้นงานจะต้องใช้โปรแกรมที่พัฒนาหลายโปรแกรมมาช่วยในการกัดชิ้นงาน อาทิเช่น การกัดรูปส่วนของทรงกลม (Sphere) ก่อนอื่นก็ต้องใช้โปรแกรมสำหรับกัดหยาบแบบ 3 แกนก่อนแล้วตามด้วยโปรแกรมสำหรับกัดละเอียดแบบ 3 แกน เป็นต้น สำหรับแบบชิ้นงานที่ใช้ทดสอบผู้วิจัยได้ใช้แบบที่มีความหลากหลาย เพื่อจะได้ทดสอบโปรแกรมได้คลอบคลุมรูปแบบการใช้งานมากที่สุด

2. ขั้นตอนการผลิตชิ้นงานจริง

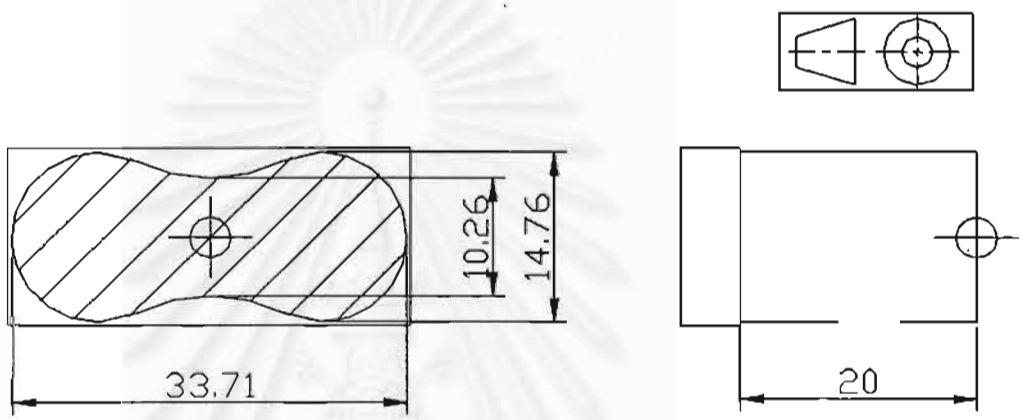
ขั้นตอนนี้ผู้วิจัยก็จะนำรหัสจี (G Code) ที่ได้จากขั้นตอนแรกไปกัดชิ้นงานจริง โดยจะใช้เครื่องกัดซีเอ็นซี (CNC Milling Machine) ของ Fanuc Model MA15 เป็นเครื่องที่ใช้ในการทดสอบรหัสจีที่ได้จากโปรแกรม ซึ่งวัสดุที่ใช้ในการทดสอบส่วนใหญ่จะใช้ไม้เป็นหลัก สาเหตุที่ใช้ไม้เป็นวัสดุในการทดสอบก็เนื่องด้วย และใช้โปรแกรม DNC ในการส่งข้อมูลรหัสจีไปยังเครื่องซีเอ็นซี

3. ขั้นตอนการทดสอบความถูกต้อง

ในการตรวจสอบความถูกต้องผู้วิจัยใช้ไมโครมิเตอร์ (Micrometer) และเครื่อง CMM ของ Brown & Sharpe ช่วยในการตรวจวัดค่าความผิดพลาดของขนาด ในส่วนการวัดขนาดของชิ้นงานที่กัดจากโปรแกรมสำหรับกัดแบบ 3 แกน จะวัดหาค่าความผิดพลาดของขนาดเฉพาะรูปทรงมาตรฐาน อาทิเช่น รูปส่วนของทรงกลมจะวัดหาค่าความผิดพลาดของค่าเส้นผ่านศูนย์กลางเป็นต้น ในการวัดขนาดความถูกต้องของชิ้นงานต้องการให้แสดงถึงความถูกต้องของการหาเส้นทางเดินของหัวกัดของโปรแกรม โดยพยายามลดสาเหตุที่ทำให้เกิดความผิดพลาดขึ้น ดังนั้นจึงไม่ได้ใช้โลหะซึ่งมีความแข็งแรงของวัสดุมากกว่าไม้ และจะเป็นเหตุให้ความแม่นยำในการกัดชิ้นงานลดลง ซึ่งในวิทยานิพนธ์นี้ใช้ไม้เป็นวัสดุก็เพื่อลดการบิดเบือนและความผิดพลาดจากกัดชิ้นงานให้น้อยลง

แบบชิ้นงานที่ 1

แบบชิ้นงานนี้ประกอบไปด้วย 2 ส่วน ส่วนแรกเป็นเส้นส่วนโค้ง (Arc) ด้านใน โดยแบบชิ้นงานนี้จะถูกเชื่อมเป็นโพลีไลน์ (Polyline) ซึ่งส่วนแรกนี้จะเป็นแบบชิ้นงานเป้าหมายของการกัด และส่วนที่ 2 เป็นเส้นตรง 4 เส้น และถูกเชื่อมเป็นโพลีไลน์ เช่นกัน โดยส่วนนี้จะเป็นส่วนที่บอกถึงขอบเขตของวัตถุดิบ (Raw Material) ในแบบชิ้นงานนี้จุดกำเนิด (Origin Point) ของแบบอยู่ตรงกลางแบบชิ้นงาน ซึ่งแสดงไว้ที่รูปที่ 4.1



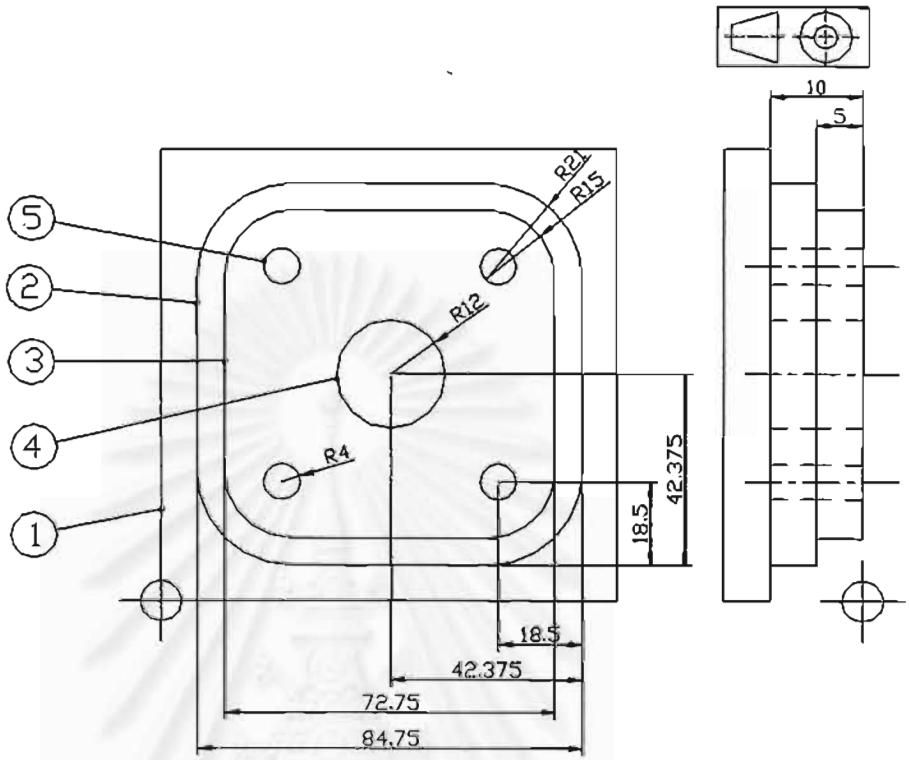
รูปที่ 4.1 ภาพแสดงแบบชิ้นงานที่ 1

ผู้วิจัยใช้โปรแกรมสำหรับงานกัดละเอียดแบบ 2 แกนครึ่งในการทดสอบแปลงแบบชิ้นงานนี้ โดยขั้นแรกก็ทำการออฟเซต (Offset) แบบชิ้นงานออก 3.5 มม. แล้วใช้โปรแกรมสำหรับงานกัดละเอียดแบบ 2 แกนครึ่งแปลงแบบชิ้นงานที่ถูกออฟเซต จากนั้นก็ออฟเซตแบบชิ้นงานเดิม ออก 3 มม. แล้วใช้โปรแกรมสำหรับงานกัดละเอียดแบบ 2 แกนครึ่งแปลงแบบชิ้นงานที่ถูกออฟเซต ทำเช่นนี้เรื่อยๆ โดยแต่ละครั้งก็ลดการออฟเซตครั้งละ 0.5 มม. จนถึงแบบชิ้นงานเดิม โดยในแต่ละครั้งก็จะกัดที่ความลึกเดียวกัน คือ 20 มม. ซึ่งจะได้ไฟล์รหัสจี 8 ไฟล์ จากนั้นก็นำไฟล์เหล่านี้ไปกัดชิ้นงานจริง โดยใช้ไม้เป็นวัตถุดิบที่ใช้ในการกัดทดสอบ และใช้หัวกัดแบบแฟลต ก็จะได้ชิ้นงานจริงตามรูปที่ 4.2 เมื่อได้ชิ้นงานจริงแล้วผู้วิจัยก็นำไปตรวจวัดด้วยไมโครมิเตอร์ ก็จะทราบค่าความผิดพลาดที่เกิดขึ้นอยู่ในช่วง 40 ถึง 60 ไมครอน โดยที่ช่วงบริเวณที่แคบที่สุดของแบบชิ้นงาน คือ 10.26 มม. จะมีค่าความผิดพลาดมากที่สุดคือ 60 ไมครอน และบริเวณอื่นจะมีค่าความผิดพลาดน้อยกว่า

รูปที่ 4.2 ภาพชิ้นงานจริงจากแบบชิ้นงานที่ 1 ที่ได้จากกัด

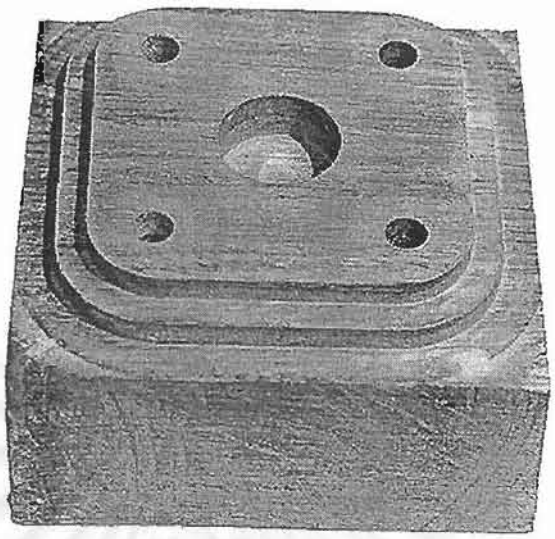
แบบชิ้นงานที่ 2

แบบชิ้นงานนี้จะถูกแบ่งออกเป็น 2 ระดับตามความสูง ดามรูปที่ 4.3 และประกอบด้วย 5 เส้นด้วยกัน คือ เส้นที่ 1 เป็นเส้นบอกขอบเขตของเนื้อวัตถุดิบ ซึ่งเป็นเส้นโพลีไลน์ เส้นที่ 2 เป็นรูปที่ต้องการกัดเป็นโพลีไลน์และจะต้องกัดลึกลงไป 10 มม. เส้นที่ 3 ก็เป็นรูปที่ต้องการกัดเป็นโพลีไลน์ เช่นกัน และกัดลึกลง 5 มม. เส้นที่ 4 เป็นรูปวงกลมตรงกลางชิ้นงาน ซึ่งรูปวงกลมนี้จะต้องคว้านเอาเนื้อด้านในวงกลมออกให้กลวง เส้นที่ 5 เป็นรูปวงกลมทั้ง 4 รูป ซึ่งจะใช้วิธีเจาะ



รูปที่ 4.3 ภาพแสดงแบบชิ้นงานที่ 2

ขั้นแรกทำการกัดหยาบก่อนโดยใช้โปรแกรมสำหรับกัดหยาบแบบ 2 แกนครึ่ง โดยกัดหยาบระหว่างเส้นที่ 3 กับเส้นที่ 1 โดยกัดลึกลงไป 5 มม. และกัดหยาบระหว่างเส้นที่ 2 กับเส้นที่ 1 โดยกัดลึกลงไป 10 มม. และกัดหยาบภายในเส้นที่ 4 ที่เป็นรูปวงกลม โดยกัดลึก 10 มม. จากนั้นก็ทำการกัดละเอียดโดยใช้โปรแกรมสำหรับกัดละเอียดแบบ 2 แกนครึ่ง โดยกัดละเอียดเส้นที่ 2, 3 และ 4 ต่อไปก็จะเป็นการเจาะรูทั้ง 4 คือเส้นที่ 5 โดยใช้โปรแกรมสำหรับงานเจาะ โดยแบ่งกัดรูละ 2 ครั้งครั้งแรกลึก 5 มม. ครั้งที่ 2 ลึก 10 มม. เมื่อดำเนินการตามขั้นตอนเหล่านี้เสร็จก็จะได้ไฟล์รหัสจีทั้งหมด 7 ไฟล์ นำไฟล์เหล่านี้ไปกัดชิ้นงานจริงด้วยเครื่องซีเอ็นซี โดยกัดตามลำดับขั้นตอนที่กล่าวมาข้างต้น และใช้ไม้เป็นวัสดุในการทดลองและหัวกัดแบบแฟลต เมื่อกัดเสร็จก็จะได้ชิ้นงานจริงตามรูปที่ 4.4 ในการทดสอบวัดนาดของชิ้นงานจริงผู้วิจัยได้ใช้เครื่อง CMM มาช่วยในการวัดขนาดสำหรับชิ้นงานนี้ โดยวัดความกว้างด้านต่าง ๆ ของชิ้นงานจริง สำหรับรูตรงกลางหรือเส้นที่ 4 ก็วัดค่าเส้นผ่านศูนย์กลางของวงกลมนั้น ส่วนเส้นที่ 5 หรือรูทั้ง 4 จะวัดค่าระยะทางระหว่างรูด้วยกันกับวัดระยะทางระหว่างรูกับรูกลวงตรงกลาง โดยวัดจากจุดศูนย์กลางรูถึงจุดศูนย์กลางรูอีกรูหนึ่ง ซึ่งผลที่ได้จากการวัดขนาดความกว้างของชิ้นงานกับค่ารัศมีของรูตรงกลางจะมีค่าความผิดพลาดอยู่ในช่วง 20 ถึง 50 ไมครอน ส่วนการวัดระยะทางระหว่างรูนั้นจะมีค่าผิดพลาดอยู่ในช่วง 10 ถึง 20 ไมครอน

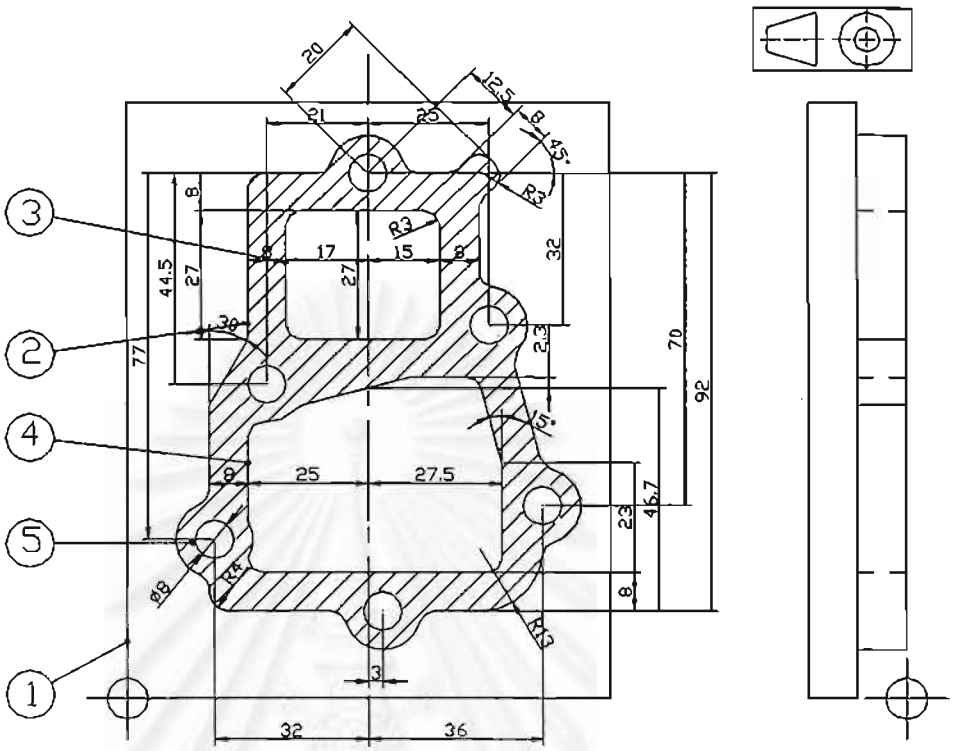


รูปที่ 4.4 ภาพชิ้นงานจริงจากแบบชิ้นงานที่ 2 ที่ได้จากกัด

แบบชิ้นงานที่ 3

แบบชิ้นงานนี้จะกัดที่ความลึกเดียวกันตลอด และประกอบไปด้วยเส้นต่างๆ จำนวน 5 เส้นด้วยกัน คือ เส้นที่ 1 ก็จะเป็นเส้นแสดงขอบเขตของวัสดุก่อนกัด ก็จะเป็นโพลีไลน์ ส่วนเส้นที่ 2 ก็เป็นเส้นขอบชิ้นงานด้านนอกที่ต้องการ เส้นที่ 3 และเส้นที่4 ก็จะเป็นเส้นขอบชิ้นงานที่ต้องด้านใน ส่วนเส้นที่ 5 จะเป็นวงกลมทั้ง 5 วง โดยแสดงแบบชิ้นงานที่ 3 ตามรูปที่ 4.5 ซึ่งส่วนที่แรเงานั้นก็คือส่วนชิ้นงานที่ต้องการหลังจากการกัด

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย



รูปที่ 4.5 ภาพแสดงแบบชิ้นงานที่ 3

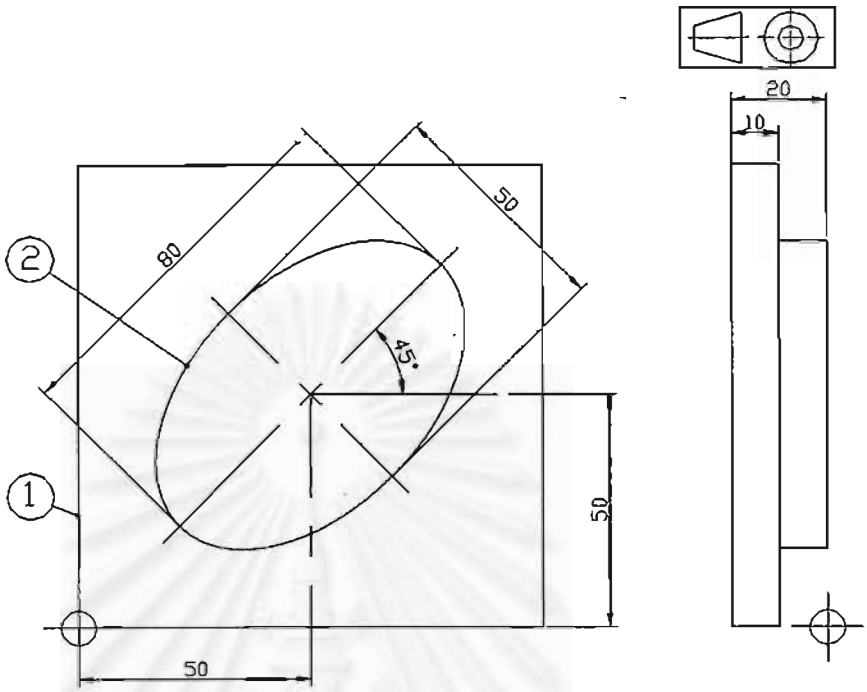
ในการกัดแบบชิ้นงานนี้ใช้โปรแกรมสำหรับงานกัดหยาบแบบ 2 แกนครึ่งก่อน โดยกัดวัสดุระหว่างเส้นที่ 1 กับเส้นที่ 2 และกัดหยาบวัสดุภายในเส้นที่ 3 และเส้นที่ 4 ก่อน แล้วจึงใช้โปรแกรมสำหรับงานกัดละเอียดแบบ 2 แกนครึ่ง โดยกัดรอบเส้นที่ 2, 3 และ 4 จากนั้นก็ใช้โปรแกรมสำหรับงานเจาะมาเจาะรูที่เส้นที่ 5 จำนวน 6 รู โดยทุกขั้นตอนจะกัดลึก 10 มม. เท่ากันหมด ต่อจากนั้นก็จะได้รหัสจีจากการใช้โปรแกรมตามข้างต้นจำนวน 7 ไฟล์ก็นำรหัสจีเหล่านี้ไปกัดชิ้นงาน โดยวัสดุที่ใช้ทดลองกัดผู้วิจัยได้เลือกใช้ไม้เป็นวัสดุในการทดลองกัด และหัวกัดแบบแฟลต เมื่อกัดชิ้นงานเสร็จแล้วก็จะได้ผลตามรูปที่ 4.6 เมื่อดูลักษณะชิ้นงานด้วยตาเปล่าก็จะเห็นได้ว่ามีรูปร่างเหมือนกับแบบชิ้นงานที่วาดขึ้น แต่เมื่อนำชิ้นงานที่ได้ไปวัดขนาดด้วยเครื่อง CMM แล้วผลที่ได้ยังมีขนาดที่ผิดจากแบบชิ้นงานที่วาดขึ้น โดยเมื่อวัดระยะจุดศูนย์กลางรูถึงรูจะมีค่าความผิดพลาดอยู่ 20 ไมครอน และวัดขนาดความกว้างของชิ้นงานมีค่าความผิดพลาดอยู่ที่ 20 ถึง 50 ไมครอน

รูปที่ 4.6 ภาพชิ้นงานจริงจากแบบชิ้นงานที่ 3 ที่ได้จากกัด

แบบชิ้นงานที่ 4

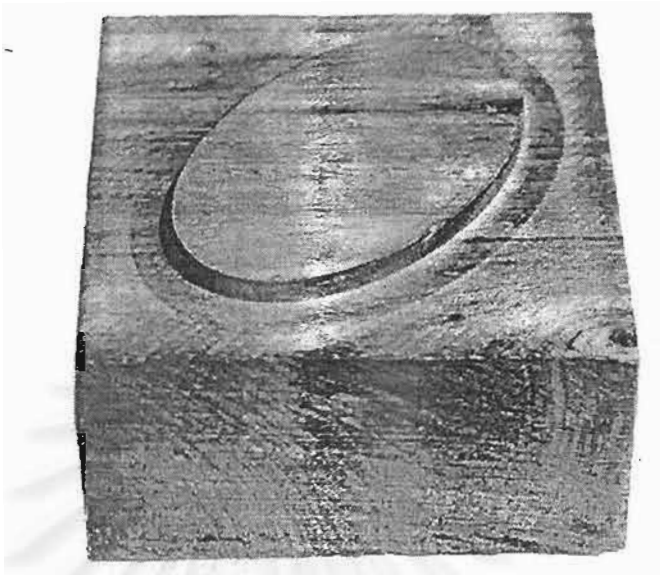
แบบชิ้นงานนี้เป็นการกัดรูปวงรีรูปเดียว แสดงที่รูปที่ 4.7 โดยกัดลึก 5 มม. ซึ่งภายในแบบจะมีเส้นแบบอยู่ 2 เส้นด้วยกันคือ เส้นที่ 1 เป็นเส้นบอกขอบเขตของวัสดุที่จะใช้ในการกัด เป็นเส้น โพลีไลน์วาดเป็นรูปสี่เหลี่ยมจัตุรัส เส้นที่ 2 เป็นเส้นวงรี ซึ่งเส้นนี้คือเส้นขอบชิ้นงานที่ต้องการ

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย



รูปที่ 4.7 ภาพแสดงแบบชิ้นงานที่ 4

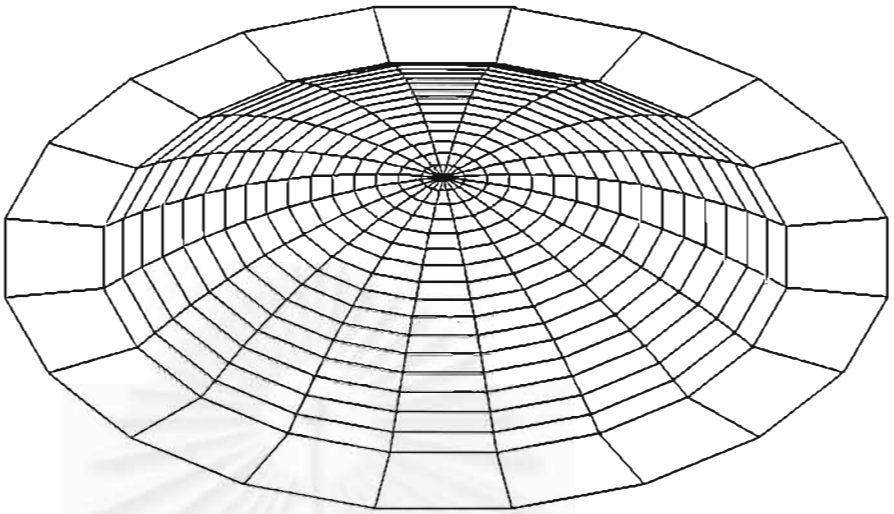
ในการกัดแบบชิ้นงานนี้จะใช้โปรแกรมสำหรับงานกัดหยาบแบบ 2 แกนครึ่ง มากัดชิ้นงานระหว่างเส้นที่ 1 กับเส้นที่ 2 จากนั้นเป็นการกัดละเอียดรอบรูปวงรีอีกครั้ง โดยใช้โปรแกรมสำหรับงานกัดละเอียดแบบ 2 แกนครึ่ง ดังนั้นก็จะได้ไฟล์รหัสจี 2 ไฟล์นำไฟล์รหัสจีส่งเข้าเครื่องกัดซีเอ็นซีเพื่อกัดงานจริง โดยวัสดุที่ใช้ทดลองกัดผู้วิจัยได้เลือกใช้ไม้เป็นวัสดุในการทดลองกัด และหัวกัดแบบแฟลต เมื่อกัดงานเสร็จแล้วก็ได้ชิ้นงานจริงตามรูปที่ 4.8 ในการตรวจวัดขนาดของชิ้นงานนี้ผู้วิจัยได้ใช้ไมโครมิเตอร์เป็นเครื่องมือในการตรวจวัด ก็เนื่องจากเครื่อง CMM ที่ใช้อยู่นั้นไม่มีฟังก์ชันในการวัดขนาดของวงรี ซึ่งผลที่ได้จากการวัดขนาดโดยรวมมีค่าความผิดพลาดของขนาดจากชิ้นงานอยู่ในช่วง 50 ถึง 70 ไมครอน



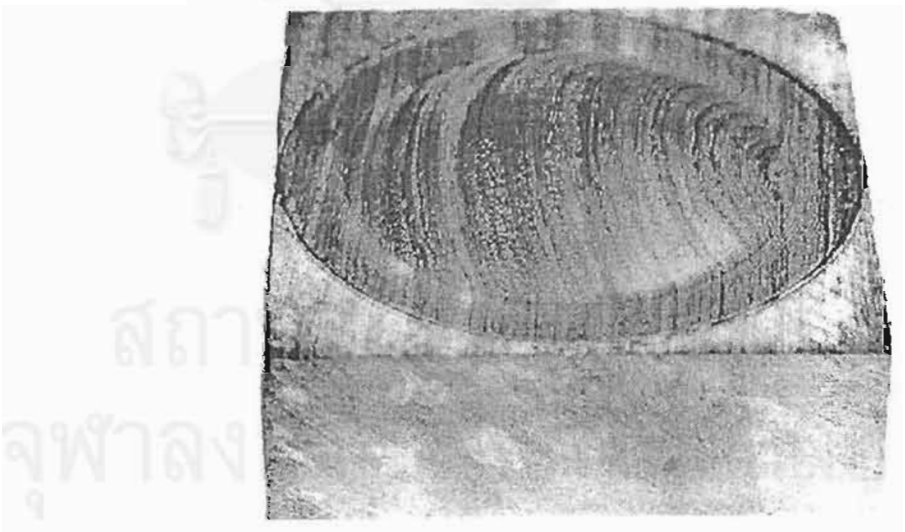
รูปที่ 4.8 ภาพชิ้นงานจริงจากแบบชิ้นงานที่ 4 ที่ได้จากกัด

แบบชิ้นงานที่ 5

แบบชิ้นงานนี้จะเป็นการกัดแบบ 3 แกน โดยแบบชิ้นงานนี้เป็นรูปส่วนของทรงกลม ซึ่งวาดจากคำสั่ง RevSurf และมีรูปผิว (Surface) เพียงผิวเดียว ซึ่งแสดงไว้ที่รูปที่ 4.9 ดังนั้นในการกัดชิ้นงานก็ใช้โปรแกรมสำหรับงานกัดหยาบแบบ 3 แกนก่อน เพื่อที่จะได้ชิ้นงานที่มีลักษณะใกล้เคียงก่อน ต่อจากนั้นก็ใช้โปรแกรมสำหรับงานกัดละเอียดแบบ 3 แกนต่อ โดยกำหนดความละเอียดในการกัดละเอียดโดยให้เส้น U Line เท่ากับ 400 เส้น และเส้น V Line เท่ากับ 500 เส้น และลักษณะการกัดจะวิ่งกัดวนรอบทรงกลม โดยวิ่งแต่ละรอบค่าพิกัด Z จะคงที่ ซึ่งในการกัดแบบชิ้นงานนี้จะได้ไฟล์รหัสจี 2 ไฟล์ เมื่อได้ไฟล์รหัสจีแล้วก็นำไฟล์ทั้ง 2 ไปกัดชิ้นงานจริง โดยวัสดุที่ใช้ก็คือไม้ ซึ่งในการกัดหยาบใช้หัวกัดแบบแฟลต (Flat End Mill) และการกัดละเอียดใช้หัวกัดแบบบอล (Ball End Mill) ผลจากการกัดก็แสดงตามรูปที่ 4.10 เมื่อนำชิ้นงานที่ได้จากการกัดจริงไปตรวจวัดขนาด ซึ่งการวัดขนาดของชิ้นงานนี้จะวัดหาค่าความผิดพลาดเส้นผ่านศูนย์กลางของทรงกลมโดยใช้เครื่อง CMM เป็นเครื่องมือในการตรวจวัดขนาด ผลที่ได้จากการวัดมีค่าความผิดพลาดอยู่ 100 ไมครอน จากรูปที่ 4.10 จะเห็นรูปเป็นร่องวงกลมที่ฐานของทรงกลมที่กัดได้เกิดจากการกัดเอาเนื้อวัสดุที่นอกเหนือจากแบบชิ้นงานที่วัดขึ้นออกด้วยมือ จึงเป็นสาเหตุให้เกิดรอยที่เห็น



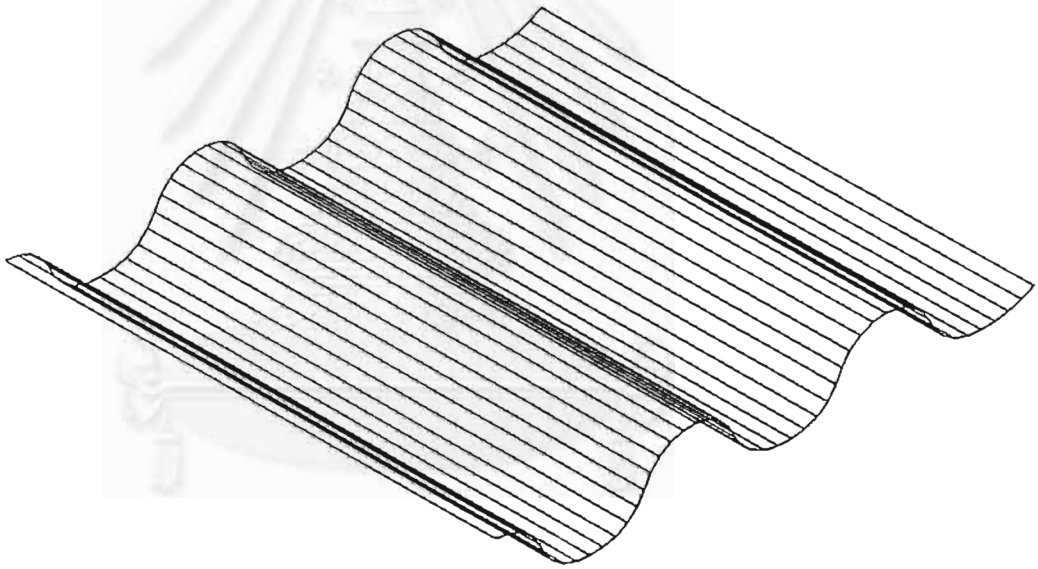
รูปที่ 4.9 ภาพแสดงแบบชิ้นงานที่ 5



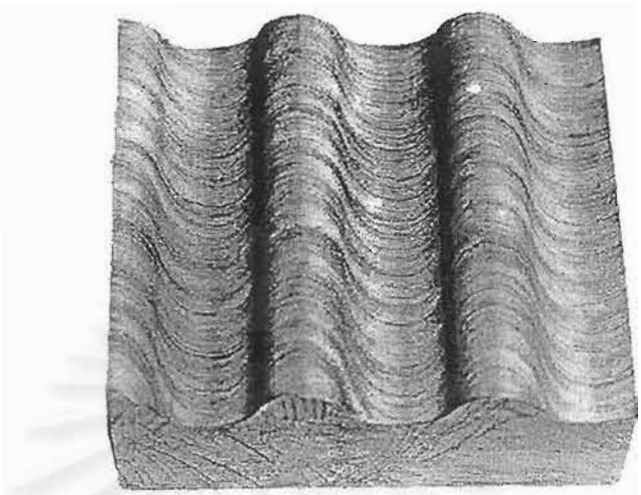
รูปที่ 4.10 ภาพชิ้นงานจริงจากแบบชิ้นงานที่ 5 ที่ได้จากกัด

แบบชิ้นงานที่ 6

แบบชิ้นงานนี้เป็นรูปลอนหรือลูกคลื่น ซึ่งเป็นรูปผิวผิวเดียวที่สร้างมาจากคำสั่ง RuleSurf ดังแสดงไว้ในรูปที่ 4.11 ในการกัดแบบชิ้นงานนี้ใช้โปรแกรมสำหรับงานกัดหยาบแบบ 3 แกนก่อน จากนั้นก็ใช้โปรแกรมสำหรับงานกัดละเอียดแบบ 3 แกน ซึ่งกำหนดความละเอียดในการกัดคือ U Line 600 เส้น และ V Line 600 เส้น โดยทิศทางการกัดจะกัดขวางลอนของแบบชิ้นงานหรือหัวกัดจะวิ่งในทิศทางตามแกน X โดยเป็นการกัดขั้นสุดท้ายเพื่อได้ชิ้นงานตามแบบที่กำหนด หลังจากนำไฟล์รหัสจีที่ได้จากโปรแกรมทั้ง 2 ไปกัดชิ้นงานจริง ในการกัดชิ้นงานได้ใช้ไม้เป็นวัตถุดิบในการทดลอง และใช้หัวกัดแบบแฟลตในการกัดหยาบ ใช้หัวกัดแบบบอลในการกัดละเอียด หลังจากกัดชิ้นงานจริงด้วยเครื่องซีเอ็นซีแล้วได้ผลตามรูปที่ 4.12 และเมื่อนำชิ้นงานไปตรวจวัดขนาดด้วยเครื่อง CMM โดยวัดหาค่าความผิดพลาดของค่ารัศมีของรูปลอนของชิ้นงาน ผลปรากฏว่ามีค่าความผิดพลาดของขนาดอยู่ที่ 115 ไมครอน



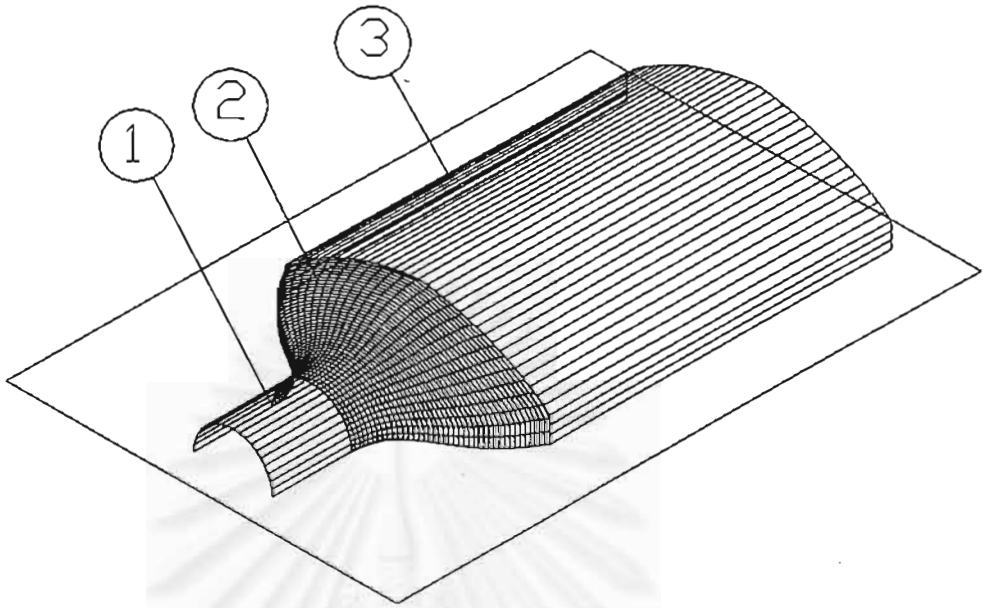
รูปที่ 4.11 ภาพแสดงแบบชิ้นงานที่ 6



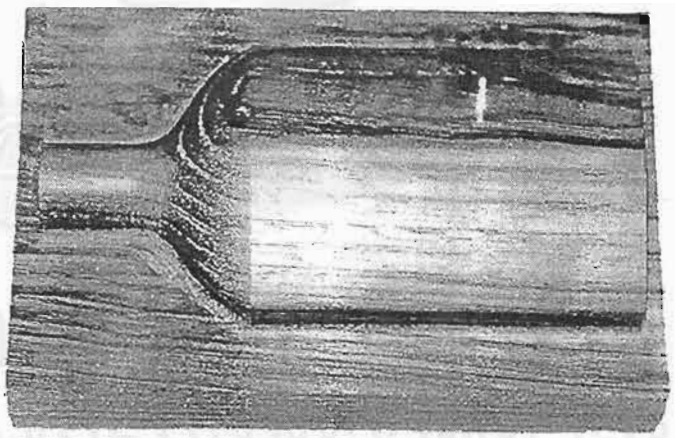
รูปที่ 4.12 ภาพชิ้นงานจริงจากแบบชิ้นงานที่ 6 ที่ได้จากกัด

แบบชิ้นงานที่ 7

แบบชิ้นงานนี้เป็นรูปขวดซึ่งประกอบไปด้วยผิวทั้งหมด 3 ผิวด้วยกัน โดยผิวชิ้นงานแรกเป็นผิวลักษณะครึ่งทรงกระบอกช่วงปากขวด ซึ่งสร้างจากคำสั่ง RuleSurf ผิวชิ้นงานที่ 2 เป็นผิวช่วงคอขวด สร้างจากคำสั่ง EdgeSurf และผิวสุดท้ายเป็นผิวช่วงตัวขวด สร้างจาก RuleSurf โดยแสดงแบบไว้ที่รูปที่ 4.13 ในการกัดแบบชิ้นงานนี้ใช้โปรแกรมสำหรับกัดหยาบแบบ 3 แกน โดยเลือกผิวเรียงจาก 1 ถึง 3 จากนั้นก็ใช้โปรแกรมสำหรับกัดละเอียดแบบ 3 แกน โดยเลือกผิวเรียงตามลำดับ และความละเอียดในการกัดของการกัดละเอียดแต่ละผิวคือ ผิวที่ 1 กำหนดให้ U Line 90 เส้น และ V Line 100 เส้น ผิวที่ 2 กำหนดให้ U Line 150 เส้น และ V Line 200 เส้น และผิวที่ 3 กำหนดให้ U Line 500 เส้น และ V Line 200 เส้น โดยทิศทางการกัดของผิวทุกผิวหัวกัดจะวิ่งในแนวแกน Y ดังนั้นจะได้ไฟล์รหัสจี 6 ไฟล์ แล้วนำไฟล์เหล่านี้ไปกัดงานจริง โดยวัสดุที่ใช้ก็คือไม้ ซึ่งในการกัดหยาบใช้หัวกัดแบบแฟลต และการกัดละเอียดใช้หัวกัดแบบบอล ผลที่ได้จากการกัดชิ้นงานจริงแสดงไว้ที่รูป 4.14 และเมื่อนำชิ้นงานที่ได้ไปวัดขนาดหาค่าความผิดพลาด โดยที่ผิวที่ 1 จะวัดค่าเส้นผ่าศูนย์กลางของปากขวด และผิวที่ 3 จะวัดขนาดความกว้างของขวด ซึ่งใช้เครื่อง CMM เป็นเครื่องมือวัด ผลที่ได้มีค่าความผิดพลาดจากของขนาดอยู่ในช่วง 105 ถึง 115 ไมครอน



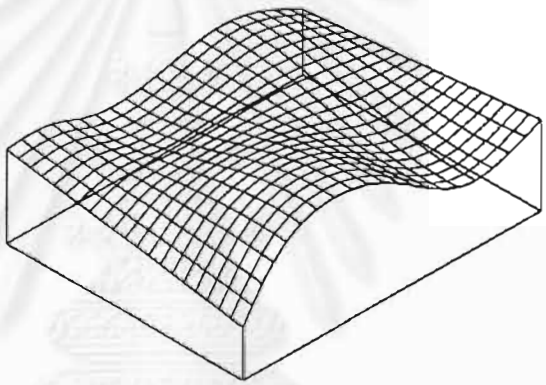
รูปที่ 4.13 ภาพแสดงแบบชิ้นงานที่ 7



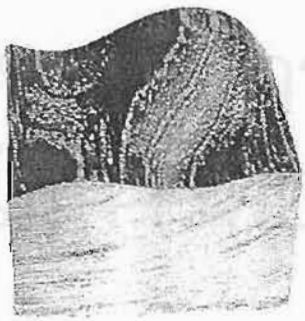
รูปที่ 4.14 ภาพชิ้นงานจริงจากแบบชิ้นงานที่ 7 ที่ได้จากกัด

แบบชิ้นงานที่ 8

แบบชิ้นงานนี้เป็นลักษณะผิวที่เรียกว่า Free Form Surface ดังแสดงไว้ในรูปที่ 4.15 ซึ่งแบบชิ้นงานนี้จะไม่ได้ทดสอบความถูกต้องของขนาด แต่จะทดสอบเรื่องลักษณะรูปร่างของการกัดจากการใช้โปรแกรมที่พัฒนา ซึ่งแบบชิ้นงานนี้จะประกอบด้วยผิวชิ้นงานเดียวที่สร้างจากคำสั่ง EdgeSurf ในการกัดชิ้นงานก็ทำการกัดหยาบก่อนโดยใช้โปรแกรมสำหรับงานกัดหยาบแบบ 3 แกน แล้วตามด้วยโปรแกรมสำหรับงานกัดละเอียดแบบ 3 แกน ก็จะได้ไฟล์รหัสจี 2 ไฟล์ ซึ่งกำหนด U Line 350 เส้น และ V Line 250 เส้น เป็นการกำหนดความละเอียดของการกัดละเอียด แล้วนำไฟล์รหัสจีไปกัดงานจริงซึ่งวัสดุที่ใช้ก็คือไม้ โดยในการกัดหยาบจะใช้หัวกัดแบบแฟลต และการกัดละเอียดใช้หัวกัดแบบบอล ซึ่งก็จะได้ผลตามรูปที่ 4.16 ซึ่งเมื่อสังเกตลักษณะรูปร่างของชิ้นงานที่ได้ก็จะเห็นได้ว่ามีรูปร่างลักษณะเหมือนกับแบบชิ้นงานที่วาดขึ้น

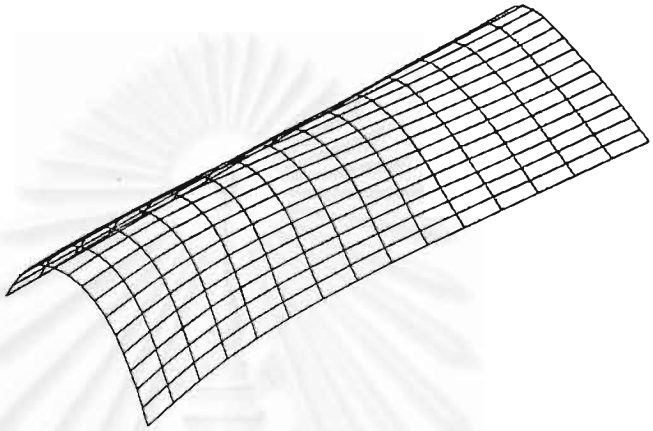


รูปที่ 4.15 ภาพแสดงแบบชิ้นงานที่ 8

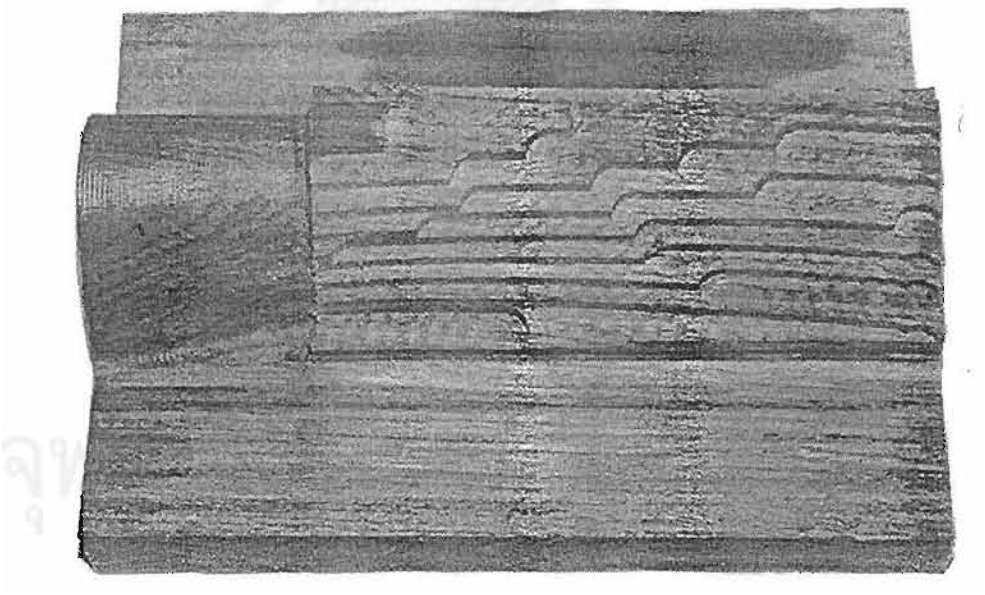


รูปที่ 4.16 ภาพชิ้นงานจริงจากแบบชิ้นงานที่ 8 ที่ได้จากกัด

และรูปที่ 4.18 เป็นภาพแสดงผลของการกัดจากโปรแกรมสำหรับงานกัดหยาบแบบ 3 แกน ซึ่งแบบชิ้นงานได้แสดงไว้ในรูปที่ 4.17 เป็นแบบของใบพัดซึ่งวาดจาก Edge Surf ในการกัดหยาบใช้หัวกัดแบบแฟลด และใช้ไม้เป็นวัสดุในการทดสอบ ซึ่งผลที่ได้ก็จะเห็นได้ว่าชิ้นงานที่ได้มีลักษณะรูปร่างคล้ายกับแบบชิ้นงาน และมีลักษณะเป็นขั้นบันได



รูปที่ 4.17 ภาพแบบชิ้นงานที่ใช้ทดสอบโปรแกรมสำหรับงานกัดหยาบแบบ 3 แกน



รูปที่ 4.18 ภาพแสดงชิ้นงานที่ถูกกัดด้วยโปรแกรมสำหรับงานกัดหยาบแบบ 3 แกน

บทที่ 5
สรุปผลการวิจัย และข้อเสนอแนะ

5.1 สรุปผลการวิจัย

สรุปผลการวิจัยโปรแกรมที่พัฒนาขึ้นทั้ง 5 โปรแกรม ผู้วิจัยได้จำแนกตามโปรแกรมดังต่อไปนี้

5.1.1 โปรแกรมสำหรับงานเจาะ

โปรแกรมสามารถหาเส้นทางเดินเพื่อไปเจาะชิ้นงานตามจุดเป้าหมายได้ และเมื่อนำชิ้นงานจริงที่ได้จากการเจาะไปวัดระยะทางระหว่างรูแล้ว ผลปรากฏว่ามีค่าความผิดพลาดอยู่ในช่วง 10 ถึง 20 ไมครอน แต่เมื่อนำพิกัดจากไฟล์รหัสจี (G Code) ที่ได้จากการใช้โปรแกรมไปเปรียบเทียบกับพิกัดที่วาดขึ้นบนโปรแกรมออโตแคด (AutoCAD) แล้วทำให้ทราบว่าค่าพิกัดในไฟล์รหัสจีที่ได้จากโปรแกรมมีค่าเท่ากับค่าพิกัดจากแบบที่วาดขึ้น นั่นแสดงว่าค่าความผิดพลาดที่เกิดขึ้นนั้นเกิดขึ้นจากขั้นตอนการกัดชิ้นงานและขั้นตอนการวัดขนาดชิ้นงาน

5.1.2 โปรแกรมสำหรับงานกัดหยาบแบบ 2 แกนครึ่ง

โปรแกรมนี้เป็นเพียงโปรแกรมใช้สำหรับกัดเนื้อวัสดุที่ไม่ต้องการออก เพื่อให้เหลือเฉพาะบริเวณที่ต้องการเท่านั้น ดังนั้นในการตรวจสอบความถูกต้องจะใช้เพียงแค่ดูลักษณะและรูปร่างของชิ้นงานที่กัดได้ ซึ่งจากผลการใช้โปรแกรมและนำไฟล์ที่ได้จากการใช้โปรแกรมไปกัดชิ้นงานจริง ผลก็คือได้ชิ้นงานที่มีลักษณะใกล้เคียงกับแบบที่วาดขึ้น แต่ว่าโปรแกรมนี้ยังมีข้อจำกัดในการใช้งานอยู่ คือ

1. เนื่องจากการหาเส้นทางเดินของหัวกัดใช้คำสั่ง Hatch จากฟังก์ชัน ARX ดังนั้นถ้าหากระยะทางระหว่างเส้นขอบเนื้อวัสดุกับเส้นขอบชิ้นงานมีค่าน้อยกว่าผลรวมของค่ารัศมีของหัวกัดกับออฟเซต (Offset) จากขอบชิ้นงาน จะทำให้การ Hatch ผิดพลาด ซึ่งจะไป Hatch ภายในเนื้อชิ้นงานแทน จึงทำให้เส้นทางเดินของหัวกัดเข้าไปกัดภายในบริเวณชิ้นงานที่ต้องการ

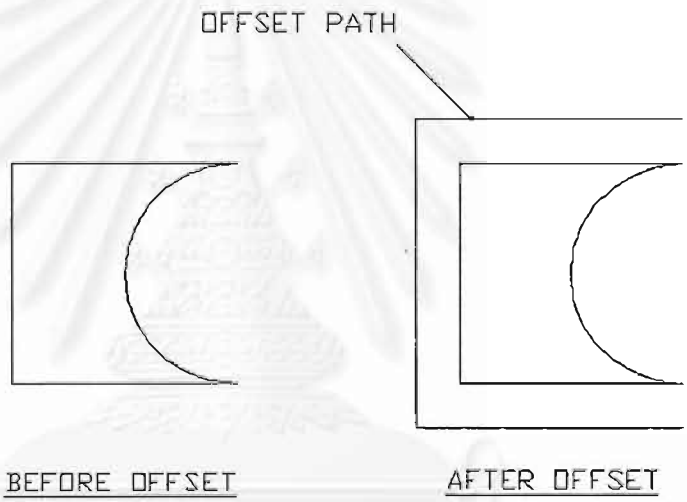
2. โปรแกรมนี้ไม่ได้มีการคำนวณหาเส้นทางเดินของหัวกัดที่ใช้เวลาน้อยที่สุด ดังนั้นเส้นทางเดินของหัวกัดที่ได้จะใช้เวลาในการกัดงานนาน และบางครั้งได้เส้นทางเดินที่ไม่เหมาะกับลักษณะการกัดแบบนั้น

5.1.3 โปรแกรมสำหรับงานกัดละเอียดแบบ 2 แกนครึ่ง

หลังจากการทดสอบโปรแกรมนี้ซึ่งได้ผลแล้วว่าชิ้นงานจริงที่กัดได้มีลักษณะเหมือนกับแบบที่วาดขึ้น และเมื่อนำชิ้นงานที่ได้ไปวัดหาค่าความผิดพลาดของขนาดผลที่ได้คือ

ค่าความผิดพลาดอยู่ในช่วง 20 ถึง 70 ไมครอน แต่เมื่อนำค่าพิกัดจากไฟล์รหัสจีที่ได้ไปเทียบกับค่าพิกัดจากแบบชิ้นงานที่ทำการเผื่อค่ารัศมีของหัวกัดแล้ว ผลปรากฏว่าเป็นค่าพิกัดเดียวกัน ดังนั้นค่าความผิดพลาดอาจจะเกิดจากขั้นตอนการกัดชิ้นงานและขั้นตอนการวัดขนาดชิ้นงาน ในการใช้งานโปรแกรมนี้มีข้อจำกัดดังต่อไปนี้

1. เนื่องจากโปรแกรมนี้ใช้ความสามารถในการคำนวณหาออฟเซตของแบบชิ้นงานจากการใช้ฟังก์ชันภายใน ARX ดังนั้นการหาออฟเซตของแบบชิ้นงานที่โปรแกรมออโตแคดไม่สามารถหาได้หรือหาได้แต่ไม่สมบูรณ์ ดังแสดงรูปตัวอย่างในรูปที่ 5.1 ซึ่งเมื่อนำแบบชิ้นงานที่ได้จากการออฟเซตไปทำการ Hatch ก็จะทำให้การ Hatch ผิดพลาดด้วย ดังนั้นจะทำให้เส้นทางการเดินของหัวกัดไม่สมบูรณ์ เมื่อกัดชิ้นงานแล้วจะทำให้ไม่ได้ชิ้นงานตามที่ต้องการ



รูปที่ 5.1 แสดงแบบชิ้นงานตัวอย่างที่ไม่สามารถออฟเซตได้สมบูรณ์

5.1.4 โปรแกรมสำหรับงานกัดหยาบแบบ 3 แกน

หลังจากใช้โปรแกรมหาเส้นทางเดินของหัวกัดได้ผลทดสอบก็คือ ชิ้นงานที่ได้มีลักษณะรูปร่างคล้ายกับแบบชิ้นงานที่ต้องการ โดยชิ้นงานที่ได้จะมีลักษณะเป็นชั้นๆ คล้ายขั้นบันได และเนื่องจากโปรแกรมนี้เป็นโปรแกรมสำหรับกัดหยาบจึงไม่ได้ทำการตรวจวัดขนาด แต่ใช้สายตาสังเกตดูลักษณะชิ้นงานเท่านั้น

5.1.5 โปรแกรมสำหรับงานกัดละเอียดแบบ 3 แกน

จากผลการทดสอบจะเห็นได้ว่ามีค่าความผิดพลาดที่วัดได้จากชิ้นงานที่กัดจริงอยู่ที่ 100 ถึง 115 ไมครอน ความผิดพลาดที่เกิดขึ้นอาจเกิดจาก ข้อจำกัดความสามารถของ ARX เช่น การสร้างรูปผิว NURBS จากจุด (Points) ที่ได้จากแบบผิวที่วาดจากโปรแกรมออโตแคด แล้วนำมาคำนวณหารูปผิวใหม่ที่เป็นรูป NURBS และอาจเกิดจากการคำนวณหารูปผิวออฟเซต NURBS จากรูปผิว NURBS และอาจเกิดจากขั้นตอนการคำนวณหาจุดที่จะใช้ในการเขียนรหัสจี ซึ่งในขั้นตอนนี้ก็จะใช้รูปผิวออฟเซต NURBS เป็นผิวในการคำนวณ หรืออาจเกิดจากขั้นตอนการกัดชิ้นงานและขั้นตอนการวัดขนาดชิ้นงาน ในการใช้งานโปรแกรมนี้มีข้อจำกัดดังต่อไปนี้

1. ไม่สามารถรับแบบชิ้นงานผิวที่แปลงมาจากโปรแกรมอื่นได้ เนื่องจากแบบชิ้นงานผิวที่แปลงมาจะไม่ได้แสดงหรือเก็บข้อมูลผิวเป็นโพลีกอนเมช (Polygonmesh)

5.2 ข้อเสนอแนะ

1. งานวิจัยในเรื่องนี้หลักสำคัญคือ การใช้ความสามารถของชุดต่อประสาน (Interface) ARX ช่วยในการคำนวณต่างๆ เช่น การคำนวณหาออฟเซตของรูปต่างๆ แต่ว่าความสามารถของ ARX ก็มีข้อจำกัดในการคำนวณอีกมาก อาทิเช่น การหาเส้นโค้ง (Curves) จากการนำเอารูปผิว (Surface) มาตัดกับระนาบ (Plane) เป็นต้น ดังนั้นจึงเป็นข้อจำกัดในการพัฒนาโปรแกรม ซึ่งในการแก้ไขก็ควรที่จะนำเอาชุด API ที่ช่วยในการคำนวณมาเพิ่มเข้าไปในโปรแกรม เพื่อช่วยเพื่อประสิทธิภาพในการทำงานของโปรแกรม

2. ถ้าหากมีการปรับปรุงโปรแกรมสำหรับงานกัดละเอียดแบบ 2 แกนครึ่ง ก็ควรจะให้มีการเลือกตำแหน่งเริ่มต้นของการกัดชิ้นงานและทิศทางของการกัดด้วย เพื่อที่จะได้ไม่ต้องเสียเวลาในการจัดลำดับการกัดขณะวาดแบบ

3. ในส่วนการจำลองเส้นทางเดินของหัวกัด (Simulation) ที่ส่วนท้ายของทุกโปรแกรม ขณะโปรแกรมกำลังจำลองเส้นทางเดินของหัวกัดควรจะ Osnap Off ก่อนมิฉะนั้นเส้นทางเดินที่ได้จากการจำลองจะผิดไปจากความเป็นจริงได้ และในส่วนเวลาที่ใช้ในการจำลองเส้นทางเดินของหัวกัดนานมากก็เนื่องมาจากโปรแกรมในส่วนนี้จะทำการอ่านค่าพิกัดจากไฟล์รหัสจีแล้วจึงนำค่าพิกัดไปจำลองเส้นทางเดินของหัวกัด นั่นก็หมายความว่าถ้าไฟล์รหัสจีใหญ่ เวลาที่ใช้ในการจำลองเส้นทางเดินของหัวกัดก็นาน ดังนั้นจึงสาเหตุทำให้โปรแกรมออโตแคดจะตัดการทำงานทุกชนิด โดยไม่รับคำสั่งใดๆ จากผู้ใช้อีกเลยจนกว่าการจำลองเส้นทางเดินของหัวกัดเสร็จ ดังนั้นในการพัฒนาโปรแกรมครั้งต่อไปก็ควรที่จะเพิ่มคุณสมบัติในการหยุดการจำลองเส้นทางเดินของหัวกัดขณะทำงานด้วย

4. ในการวาดรูปผิวควรจะกำหนดจำนวนเส้นในแนวแกน U และ V ที่ใช้ในการแสดงรูปผิวให้เหมาะสมไม่มากน้อยจนเกินไป ก็เพราะถ้ามากก็จะทำให้ใช้เวลาในการคำนวณหา NURBS Surface มาก แต่ว่าก็จะทำให้ความถูกต้องของรูปผิว NURBS สูงขึ้น ถ้าน้อยไปก็จะทำให้ความถูกต้องของรูปผิว NURBS น้อยลง แต่ว่าจะใช้เวลาในการคำนวณหารูปผิว NURBS ลดลง

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

รายการอ้างอิง

ภาษาไทย

ชาลี ตระการกูล. เทคโนโลยีซีเอ็นซี. พิมพ์ครั้งที่ 1. กรุงเทพมหานคร: สมาคมส่งเสริมเทคโนโลยี (ไทย- ญี่ปุ่น), 2537.

สาโรช พรวิจิตรจินดา. การประยุกต์ใช้คอมพิวเตอร์กราฟฟิกส์กับการควบคุมการทำงานของเครื่องซีเอ็นซี. วิทยานิพนธ์ปริญญามหาบัณฑิต ภาควิชาวิศวกรรมเครื่องกล คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย, 2531.

ศุภชัย รมยานนท์ และ ฉวีวรรณ รมยานนท์. ทฤษฎีงานเครื่องมือกลเบื้องต้น งานกัด. พิมพ์ครั้งที่ 1. กรุงเทพมหานคร: ไทยวัฒนาพานิช, 2529.

ภาษาอังกฤษ

AutoCAD Market Group. AutoCAD Software Development Kit (SDK). USA: Autodesk, 1997.

Farid M.L. Amirouche. Computer-Aided Design and Manufacturing. 1st ed. USA: Prentice Hall, 1993.

Micheal D. Stewart. Modeling for Design Using AutoCAD Release13 and AutoSurf. 1st ed. USA: Autodesk Press, 1997.

Ori Gurewich, and Nathan Gurewich. Teach Yourself Visual C++ 5 in 21 Days. 4th ed. USA: Sams Publishing, 1997.

Owen Ransen. AutoCAD programming in C/C++. 1st ed. USA: John Wiley & Son, 1997.

Rusty Gesner. Maximizing AutoCAD R13. 1st ed. USA: Autodesk Press, 1997.

S.C. Jonathan Lin. Computer Numerical Control From Programming to Networking. 1st ed. USA: Delmar, 1994.

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

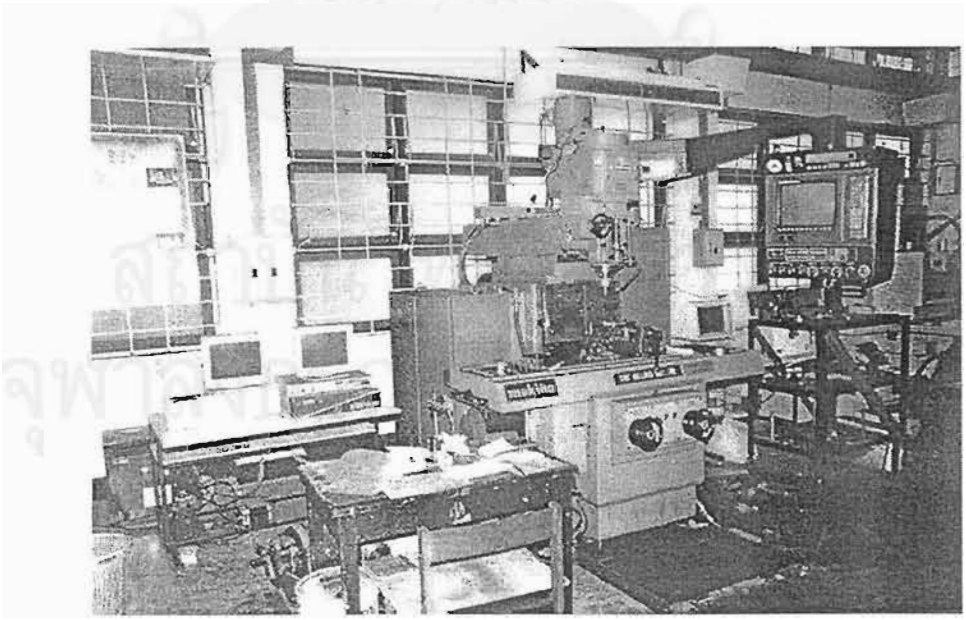
ภาคผนวก

ภาคผนวก ก.
อุปกรณ์และเครื่องมือที่ใช้ในวิทยานิพนธ์

ก.1 เครื่องกัดซีเอ็นซี (CNC Milling Machine)

เครื่องกัดซีเอ็นซีที่ใช้ในวิทยานิพนธ์นี้เป็นของ Makino และชุดควบคุม (Controller) ที่ใช้เป็นของ Fanuc Model MA15 ซึ่งสามารถกัดชิ้นงานในลักษณะ 4 แกนได้ โดยแกนที่ 4 จะเป็นการเพิ่มการวางทิศทาง (Orientation) ของแกน X ซึ่งเป็นชุดอุปกรณ์เพิ่มวางบนแท่นกัดชิ้นงานของเครื่องซีเอ็นซี แต่ในวิทยานิพนธ์นี้ใช้ความสามารถในการกัดชิ้นงานแค่ 3 แกนเท่านั้น ดังนั้นแกนที่ 4 จึงไม่ได้ใช้และยกออกจากแท่นกัดชิ้นงาน ได้แสดงเครื่องซีเอ็นซีที่ใช้ในรูปที่ ก.1 และรายละเอียดของเครื่องซีเอ็นซีรุ่นนี้มีดังต่อไปนี้

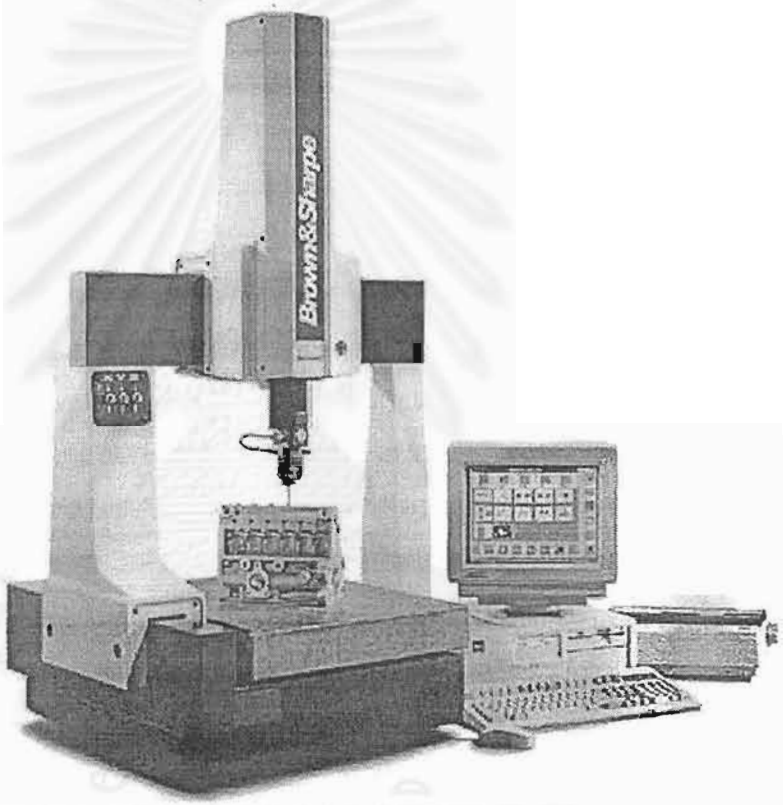
Manufacturer	: Makino
Model	: AVNC 40
Controller	: FANUC Model MA15
Table Size	: 1050x400 mm (710x400 mm)
Vertical Head Travel	: 360 mm
Spindle Speed	: 250 – 4000 rpm
Spindle Nose	: ISO NO.40
Accuracy	: 0.020 mm.



รูปที่ ก.1 ภาพแสดงเครื่องกัดซีเอ็นซีที่ใช้ในการกัดชิ้นงาน

ก.2 เครื่องมือวัดพิกัด (Coordinate Measuring Machine, CMM)

เครื่อง CMM ที่ใช้ในวิทยานิพนธ์เป็นของ Brown & Sharpe รุ่น MicroVal 343 และได้แสดงรูปชุดเครื่องมือวัดพิกัดที่ใช้ไว้ในรูปที่ ก.2 โดยการทำงานจะใช้ซอฟต์แวร์ชื่อ MicroMeasure three version 6.1 เป็นซอฟต์แวร์ในการคำนวณหาพิกัด ในการตรวจวัดพิกัดจะใช้หัวตรวจ (Probe) ที่ทำมาจากทับทิมมาสัมผัสกับชิ้นงานที่ต้องการวัดพิกัด และรายละเอียดข้อกำหนดเครื่องแสดงไว้ในตารางที่ ก.1



รูปที่ ก.2 ภาพแสดงเครื่องมือวัดพิกัดของ Brown & Sharpe รุ่น MicroVal 343

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

PERFORMANCE	METRIC	ENGLISH
Repeatability	0.004 mm	0.00016 in.
Volumetric Accuracy	0.010 mm	0.0004 in.
Linear Accuracy	0.005 mm	0.0002 in.
Resolution	0.0005 mm	0.00002 in.
Display Range	+/-9999.999 mm	+/-999.99999 in.
Measuring Speed (Min.)	760 mm/sec	30 in/sec
DIMENSIONS		
Measuring Range	356x406x305 mm	14x16x12 in.
Length	753 mm	29.25 in.
Width	730 mm	28.75 in.
Height	1340 mm	52.75 in.
Weight (Machine/granite)	150 kg	330 lbs
Weight (Complete system)	168 kg	370 lbs
Shipping Weight	234 kg	515 lbs
Maximum Part Weight	227 kg	500 lbs
Part Size Capability (X, Y, Z)	457x610x381 mm	18x24x15 in.
OPERATIONAL REQUIREMENTS		
Calibration Temperature	20°C+/-1.1°C	68°F+/-2°F
Operating Temp. Range	10° to 40°C	50° to 104°F
Storage Temperature	-30° to 60°C	-22° to 140°F
Minimum Air Input	4.8 BAR	70 psi
Air Consumption	100NL@3.75BAR	3.5SCFM@55psi
Power Requirements	100 to 240 VAC 50/60HZ	100 to 240 VAC 50/60HZ
Power Consumption-Elec.	200 Watts Max.	200 Watts Max.
Power Consumption-CRT	50 Watts Max.	50 Watts Max.

ตารางที่ ก.1 แสดงรายละเอียดข้อกำหนดเครื่องมือวัดพิกัดของ Brown & Sharpe รุ่น MicroVal 343

ก.3 ไมโครมิเตอร์ (Micrometer)

ไมโครมิเตอร์ที่ใช้มีอยู่ 2 แบบ คือ 1) แบบปลายแฟลต (Flat) 2) แบบปลายบอล (Ball) โดยแบบปลายบอลที่ใช้มีอยู่ขนาดเดียวคือ วัดขนาดได้ในช่วง 0-25 มม. ซึ่งแสดงไว้ในรูปที่ ก.3 ส่วนแบบปลายแฟลดที่ใช้มีอยู่ 2 ขนาด คือ ขนาดแรกจะวัดได้ในช่วง 0-25 มม. ได้แสดงไว้ในรูปที่ ก.4 ขนาดที่ 2 จะวัดได้ในช่วง 25-50 มม. แสดงไว้ในรูปที่ ก.5 ไมโครมิเตอร์ทั้งหมดที่ใช้อยู่เป็นของ Mitutoyo โดยวัดได้ละเอียดมากที่สุด 10 ไมครอน

รูปที่ ก.3 แสดงภาพไมโครมิเตอร์แบบปลายบอล

รูปที่ ก.4 แสดงภาพไมโครมิเตอร์แบบปลายแฟลตวัดได้ในช่วง 0-25 มม.

รูปที่ ก.5 แสดงภาพไมโครมิเตอร์แบบปลายแฟลตวัดได้ในช่วง 25-50 มม.

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

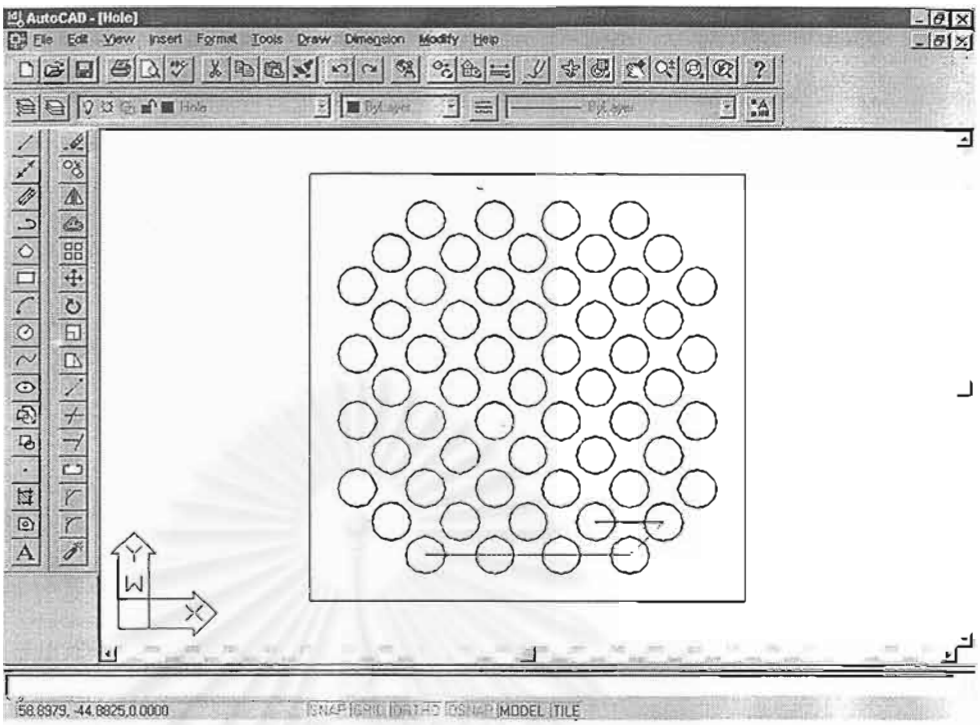
ก.4 หัวกัด (End Mill)

หัวกัดที่ใช้มีอยู่ 2 แบบ คือ แบบหัวแฟลดและแบบหัวบอล โดยหัวกัดแบบแฟลดใช้ขนาดเส้นผ่านศูนย์กลาง 8 มม. ซึ่งเป็นหัวไททาเนียม เป็นของยี่ห้อ YG ส่วนหัวกัดแบบบอลใช้ขนาดเส้นผ่านศูนย์กลาง 2 มม. ซึ่งเป็นหัวไฮสปีด (High Speed) ซึ่งหัวกัดทั้ง 2 แบบแสดงไว้ในรูปที่ ก.6

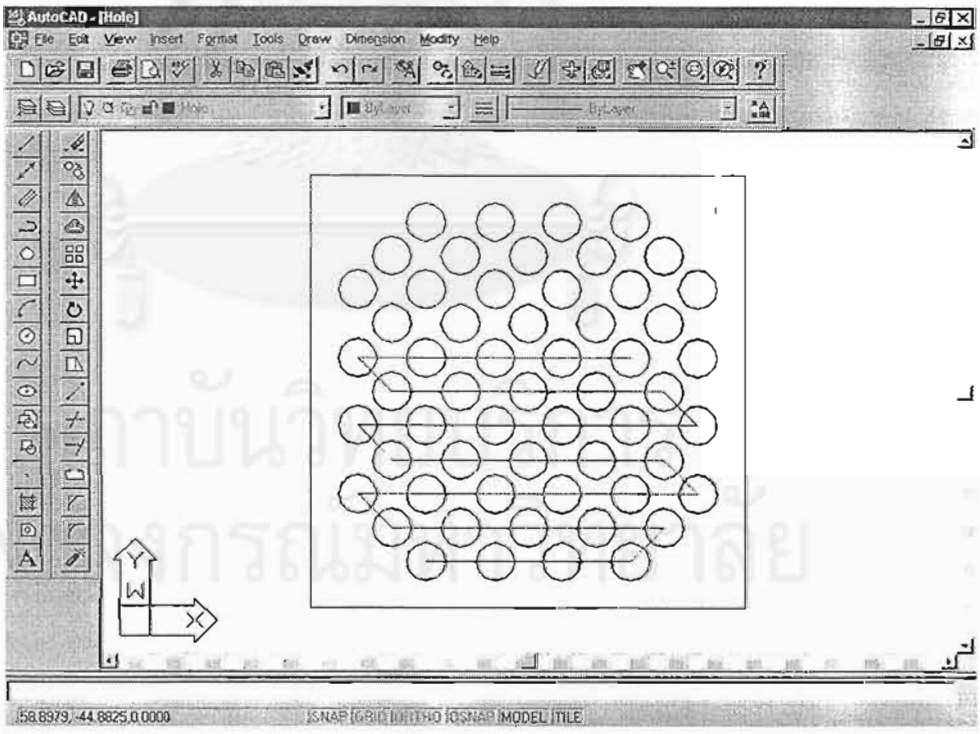
รูปที่ ก.6 ภาพแสดงหัวกัดที่ใช้

ภาคผนวก ข.
การจำลองเส้นทางเดินของหัวกัด

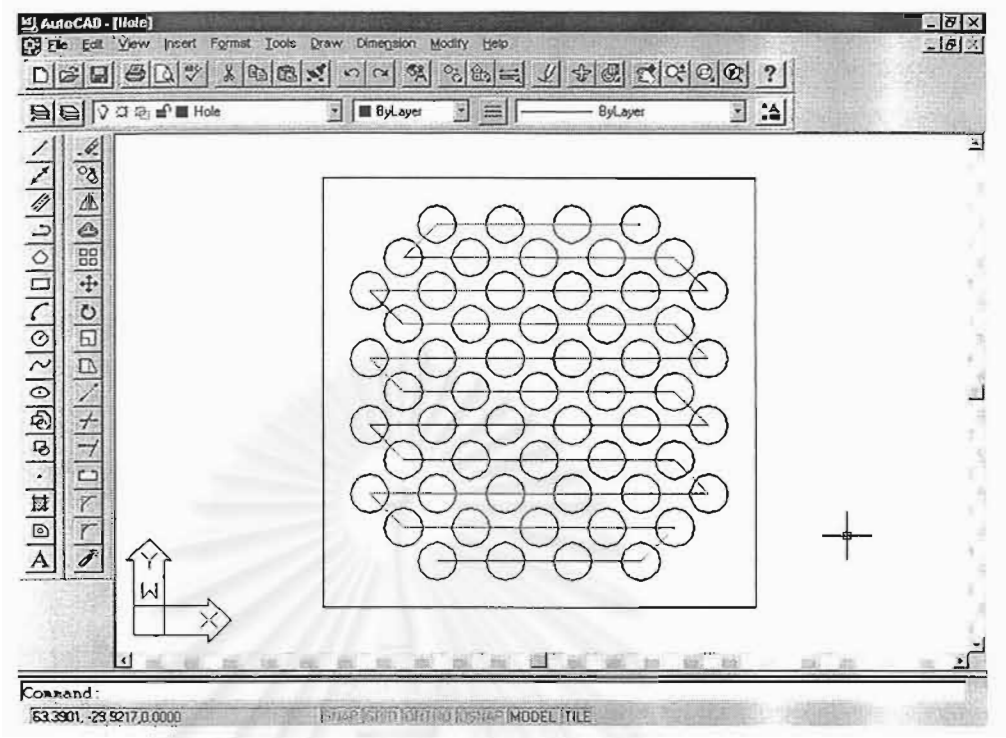
การจำลองเส้นทางเดินของหัวกัดจะมีอยู่ส่วนท้ายของทุกโปรแกรมที่พัฒนาขึ้น ซึ่งหลักการทำงานก็คืออ่านไฟล์รหัสจีที่โปรแกรมเขียนขึ้น แล้วนำมาวาดเป็นเส้นตรงเมื่ออ่านพบ G01 และนำมาวาดเป็นเส้นโค้งเมื่ออ่านพบ G02 หรือ G03 จากหลักการดังกล่าวคือเป็นการไฟล์รหัสจีโดยตรงดังนั้นถ้าหากไฟล์รหัสจีมีขนาดใหญ่ก็จะทำให้การจำลองเส้นทางเดินของหัวกัดใช้เวลานาน และในภาคผนวกนี้จะแสดงภาพการจำลองเส้นทางเดินของหัวกัดของทุกโปรแกรมที่พัฒนาขึ้น โดยภาพแรกที่แสดงไว้ที่รูปที่ ข.1 เป็นภาพการจำลองเส้นทางเดินของหัวกัดของโปรแกรมสำหรับงานเจาะขณะเริ่มต้น รูปที่ ข.2 เป็นภาพการจำลองเส้นทางเดินของหัวกัดของโปรแกรมสำหรับงานเจาะขณะกำลังกัดชิ้นงาน รูปที่ ข.3 เป็นภาพการจำลองเส้นทางเดินของหัวกัดของโปรแกรมสำหรับงานเจาะเมื่อจบจำลองเส้นทางเดินของหัวกัด รูปที่ ข.4 เป็นภาพการจำลองเส้นทางเดินของหัวกัดของโปรแกรมสำหรับงานกัดหยาบแบบ 2 แกนครึ่งขณะเริ่มต้น รูปที่ ข.5 เป็นภาพการจำลองเส้นทางเดินของหัวกัดของโปรแกรมสำหรับงานกัดหยาบแบบ 2 แกนครึ่งขณะกำลังกัดชิ้นงาน รูปที่ ข.6 เป็นภาพการจำลองเส้นทางเดินของหัวกัดของโปรแกรมสำหรับงานกัดหยาบแบบ 2 แกนครึ่งเมื่อจบจำลองเส้นทางเดินของหัวกัด รูปที่ ข.7 เป็นภาพการจำลองเส้นทางเดินของหัวกัดของโปรแกรมสำหรับงานกัดละเอียดแบบ 2 แกนครึ่งขณะเริ่มต้น รูปที่ ข.8 เป็นภาพการจำลองเส้นทางเดินของหัวกัดของโปรแกรมสำหรับงานกัดละเอียดแบบ 2 แกนครึ่งขณะกำลังกัดชิ้นงาน รูปที่ ข.9 เป็นภาพการจำลองเส้นทางเดินของหัวกัดของโปรแกรมสำหรับงานกัดละเอียดแบบ 2 แกนครึ่งเมื่อจบจำลองเส้นทางเดินของหัวกัด รูปที่ ข.10 เป็นภาพการจำลองเส้นทางเดินของหัวกัดของโปรแกรมสำหรับงานกัดหยาบแบบ 3 แกนขณะเริ่มต้น รูปที่ ข.11 เป็นภาพการจำลองเส้นทางเดินของหัวกัดของโปรแกรมสำหรับงานกัดหยาบแบบ 3 แกนขณะกำลังกัดชิ้นงาน รูปที่ ข.12 เป็นภาพการจำลองเส้นทางเดินของหัวกัดของโปรแกรมสำหรับงานกัดหยาบแบบ 3 แกนเมื่อจบจำลองเส้นทางเดินของหัวกัด รูปที่ ข.13 เป็นภาพการจำลองเส้นทางเดินของหัวกัดของโปรแกรมสำหรับงานกัดละเอียดแบบ 3 แกนขณะเริ่มต้น รูปที่ ข.14 เป็นภาพการจำลองเส้นทางเดินของหัวกัดของโปรแกรมสำหรับงานกัดละเอียดแบบ 3 แกนขณะกำลังกัดชิ้นงาน และรูปที่ ข.15 เป็นภาพการจำลองเส้นทางเดินของหัวกัดของโปรแกรมสำหรับงานกัดละเอียดแบบ 3 แกนเมื่อจบจำลองเส้นทางเดินของหัวกัด โดยตั้งแต่รูปที่ ข.12 ถึงรูปที่ ข.15 จะแสดงรูปแบบการกัดละเอียดแบบ One-Way ส่วนตั้งแต่รูปที่ ข.16 ถึงรูปที่ ข.18 จะเป็นภาพการจำลองเส้นทางเดินของหัวกัดของโปรแกรมสำหรับงานกัดละเอียดแบบ 3 แกนเช่นแต่จะแสดงรูปแบบการกัดละเอียดแบบ Zig-Zag



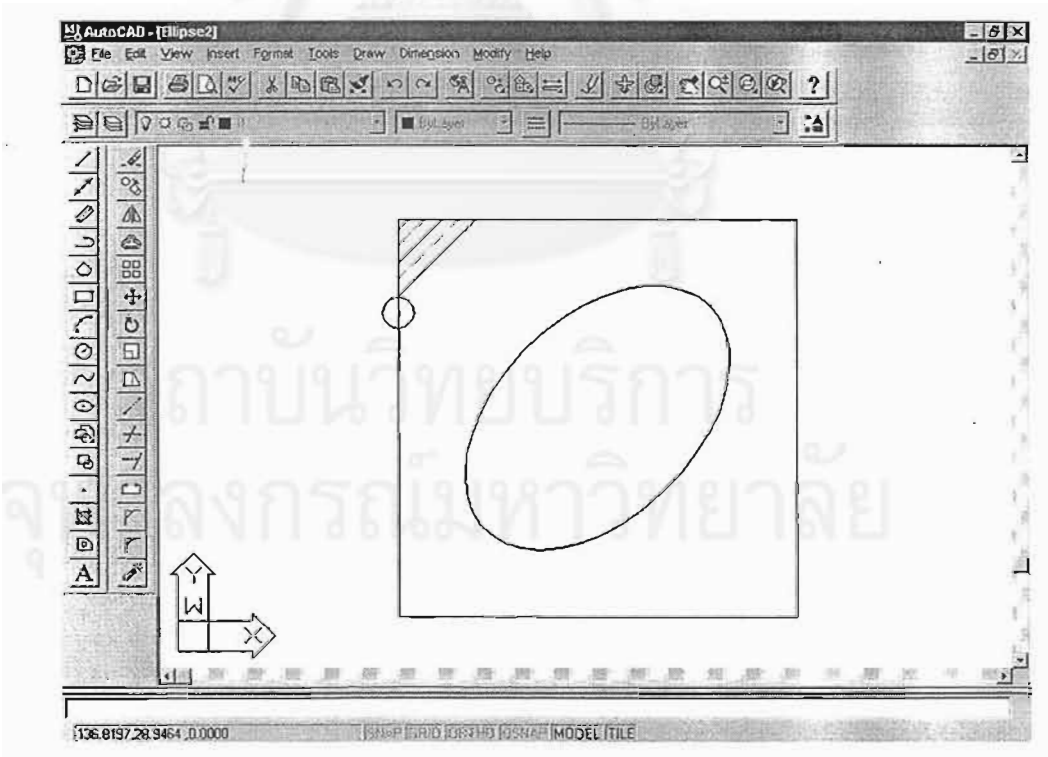
รูปที่ ข.1 ภาพการจำลองเส้นทางเดินของหัวกัดของโปรแกรมสำหรับงานเจาะขณะเริ่มตัน



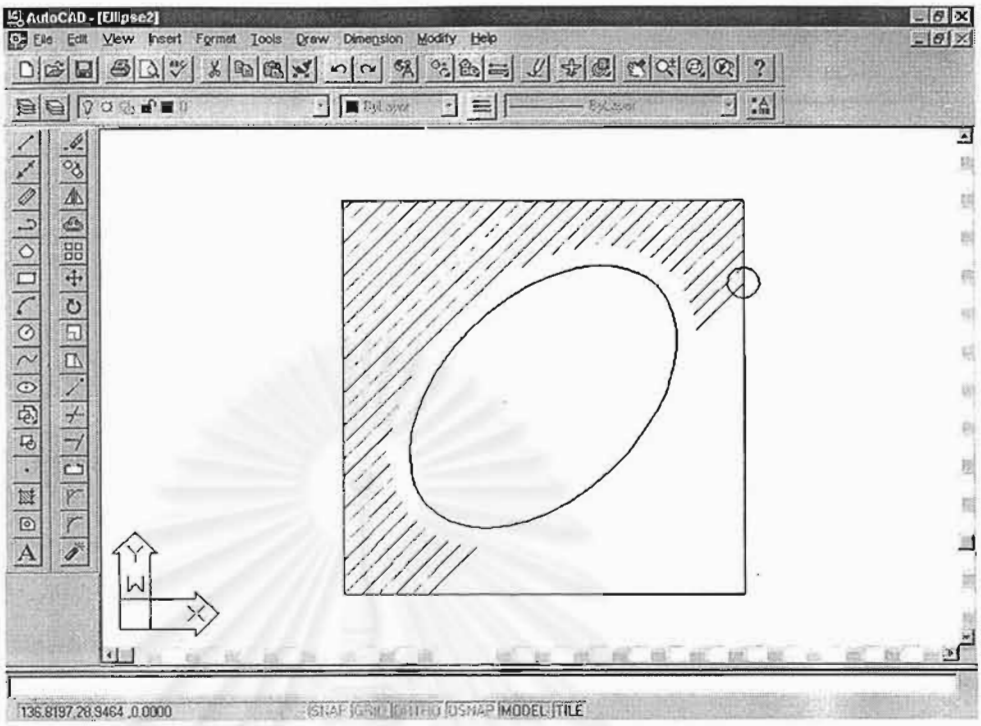
รูปที่ ข.2 ภาพการจำลองเส้นทางเดินของหัวกัดของโปรแกรมสำหรับงานเจาะ
ขณะกำลังกัดชิ้นงาน



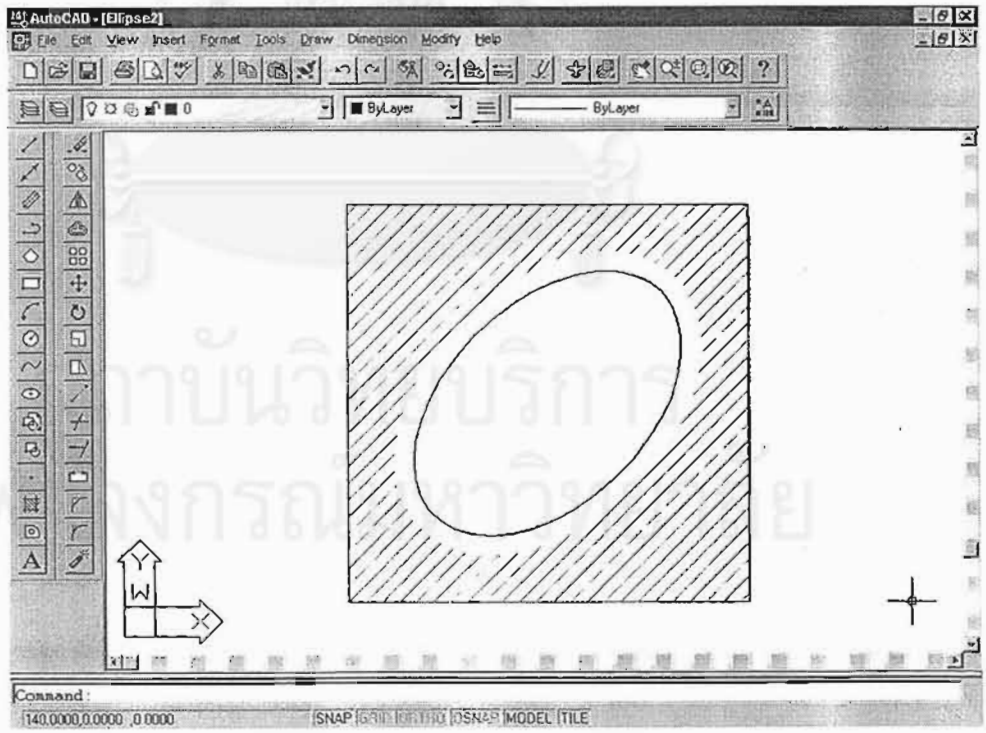
รูปที่ ข.3 ภาพการจำลองเส้นทางเดินของหัวกัดของโปรแกรมสำหรับงานเจาะเมื่อจบจำลองเส้นทางเดินของหัวกัด



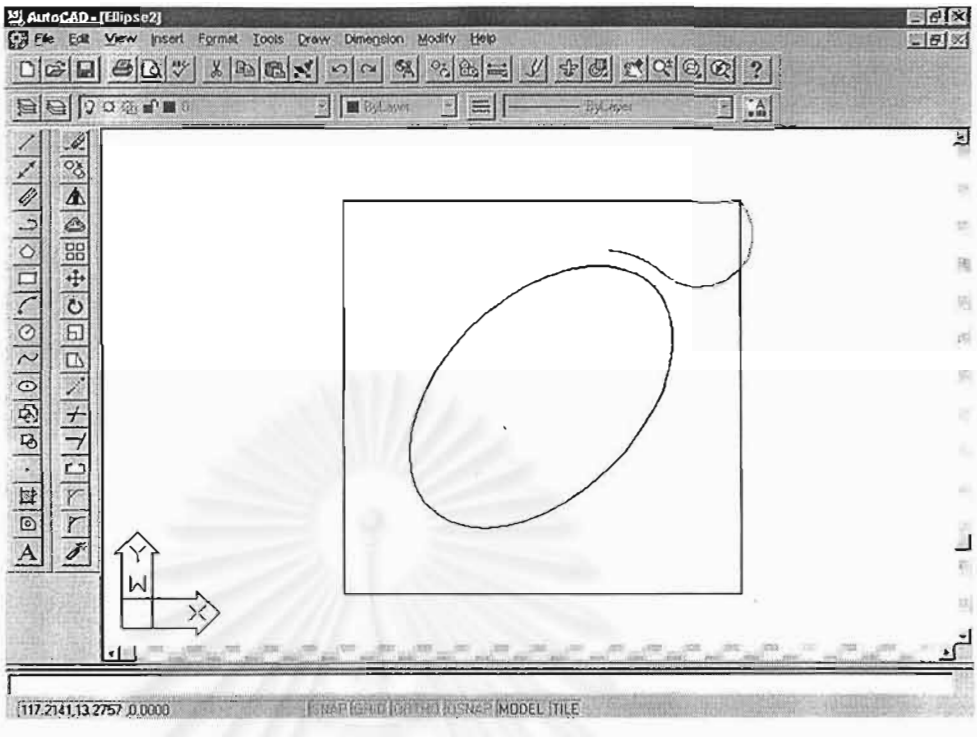
รูปที่ ข.4 ภาพการจำลองเส้นทางเดินของหัวกัดของโปรแกรมสำหรับงานกัดหยาบแบบ 2 แกนครึ่งขณะเริ่มต้น



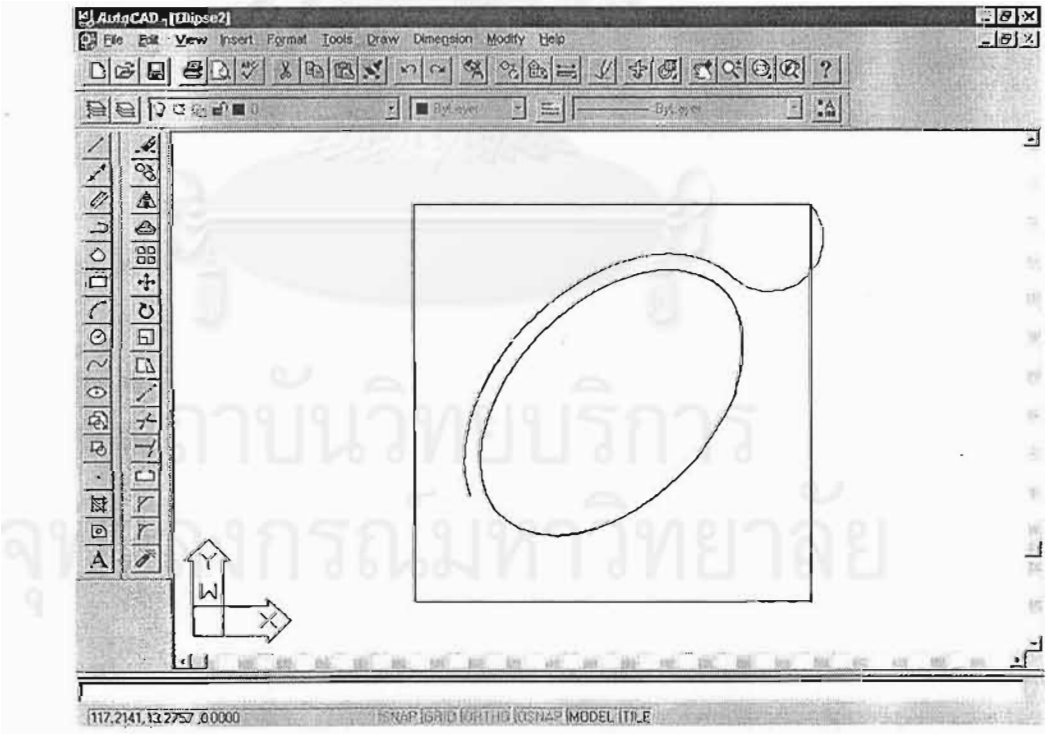
รูปที่ ข.5 ภาพการจำลองเส้นทางเดินของหัวกัดของโปรแกรมสำหรับงานกัดหยาบแบบ 2 แกนครึ่งขณะกำลังกัดชิ้นงาน



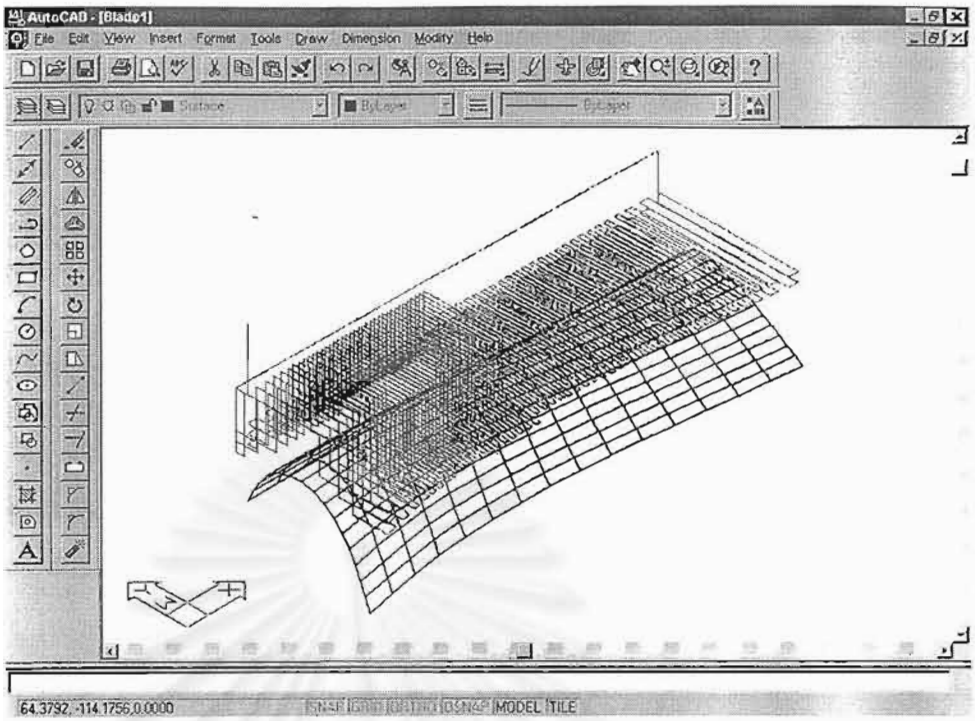
รูปที่ ข.6 ภาพการจำลองเส้นทางเดินของหัวกัดของโปรแกรมสำหรับงานกัดหยาบแบบ 2 แกนครึ่งเมื่อจบจำลองเส้นทางเดินของหัวกัด



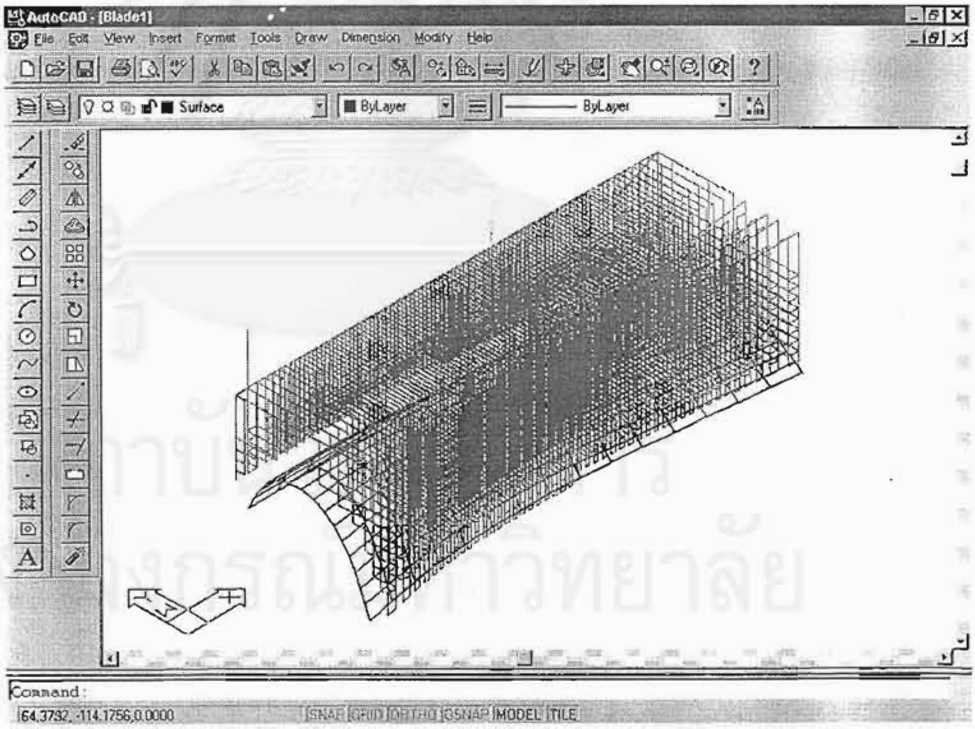
รูปที่ ข.7 ภาพการจำลองเส้นทางเดินของหัวกัดของโปรแกรมสำหรับงานกัดละเอียดแบบ 2 แกนครึ่งขณะเริ่มต้น



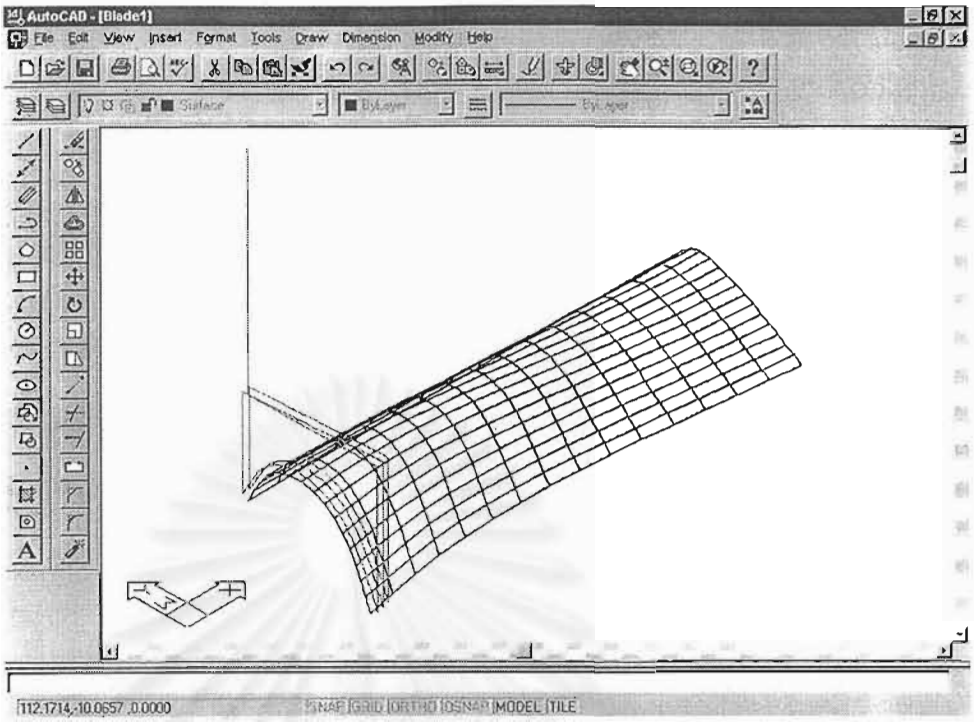
รูปที่ ข.8 ภาพการจำลองเส้นทางเดินของหัวกัดของโปรแกรมสำหรับงานกัดละเอียดแบบ 2 แกนครึ่งขณะกำลังกัดชิ้นงาน



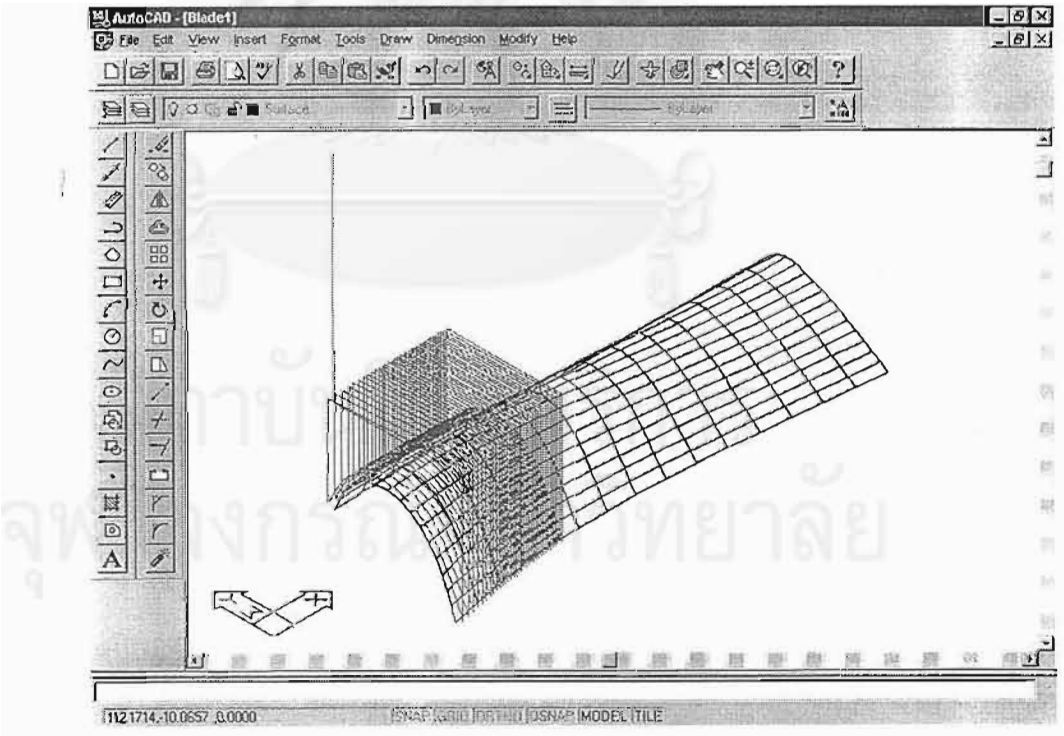
รูปที่ ข.11 ภาพการจำลองเส้นทางเดินของหัวกัดของโปรแกรมสำหรับงานกัดหยาบแบบ 3 แกนขณะกำลังกัดชิ้นงาน



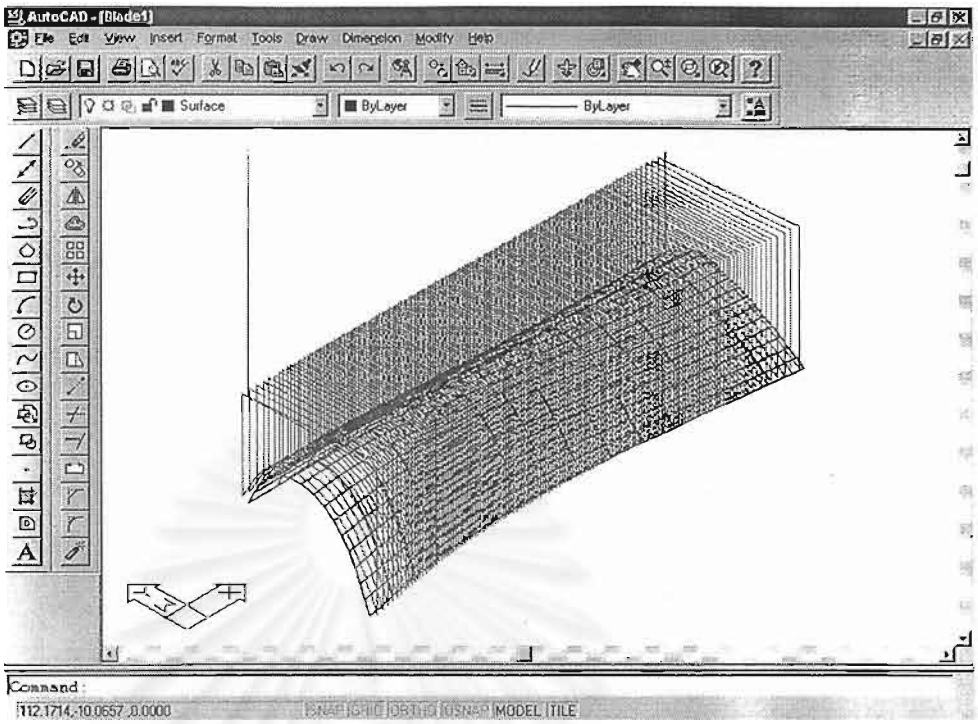
รูปที่ ข.12 ภาพการจำลองเส้นทางเดินของหัวกัดของโปรแกรมสำหรับงานกัดหยาบแบบ 3 แกนเมื่อจบจำลองเส้นทางเดินของหัวกัด



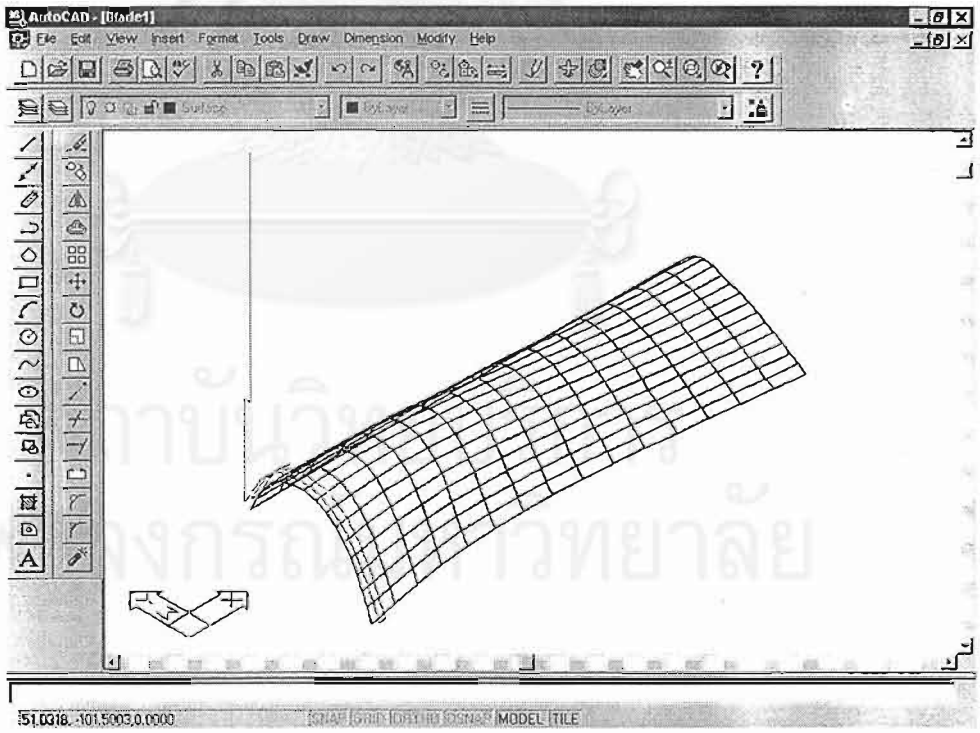
รูปที่ ข.13 ภาพการจำลองเส้นทางเดินของหัวกัดของโปรแกรมสำหรับงานกัดละเอียดแบบ 3 แกน วิ่งแบบ One-Way ขณะเริ่มตัน



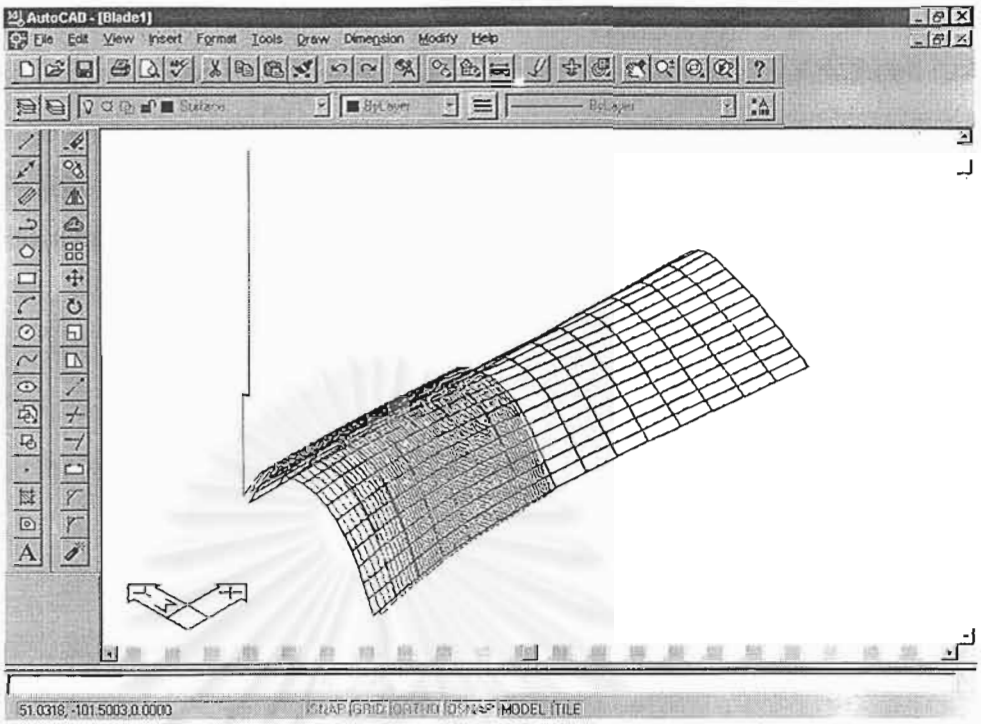
รูปที่ ข.14 ภาพการจำลองเส้นทางเดินของหัวกัดของโปรแกรมสำหรับงานกัดละเอียดแบบ 3 แกน วิ่งแบบ One-Way ขณะกำลังกัดชิ้นงาน



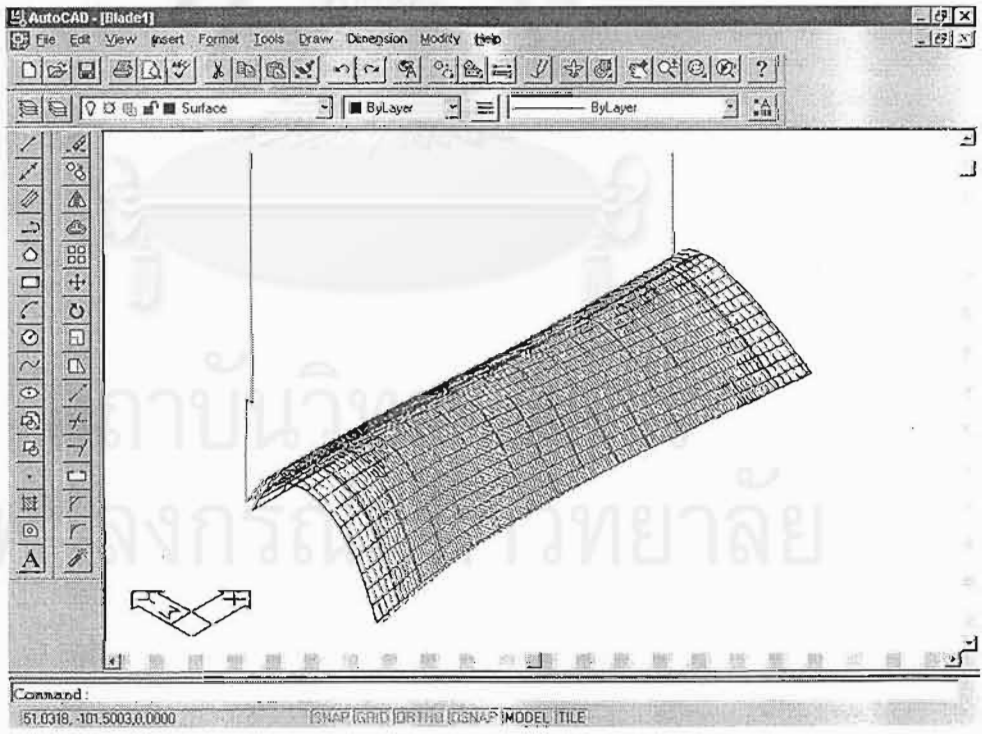
รูปที่ ข.15 ภาพการจำลองเส้นทางเดินของหัวกัดของโปรแกรมสำหรับงานกัดละเอียดแบบ 3 แกน วิ่งแบบ One-Way เมื่อจบจำลองเส้นทางเดินของหัวกัด



รูปที่ ข.16 ภาพการจำลองเส้นทางเดินของหัวกัดของโปรแกรมสำหรับงานกัดละเอียดแบบ 3 แกน วิ่งแบบ Zig-Zag ขณะเริ่มต้น



รูปที่ ข.17 ภาพการจำลองเส้นทางเดินของหัวกัดของโปรแกรมสำหรับงานกัดละเอียดแบบ 3 แกน วิ่งแบบ Zig-Zag ขณะกำลังกัดชิ้นงาน



รูปที่ ข.18 ภาพการจำลองเส้นทางเดินของหัวกัดของโปรแกรมสำหรับงานกัดละเอียดแบบ 3 แกน วิ่งแบบ Zig-Zag เมื่อจบจำลองเส้นทางเดินของหัวกัด

ประวัติผู้เขียน

นายมนูศักดิ์ จานทอง เกิดวันที่ 2 มีนาคม พ.ศ. 2516 ที่อำเภอพนมทวน จังหวัดกาญจนบุรี สำเร็จการศึกษาปริญญาตรีวิศวกรรมศาสตร์บัณฑิต ภาควิชาวิศวกรรมเครื่องกล คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีมหานคร ในปีการศึกษา 2538 และเข้าศึกษาต่อในหลักสูตรวิศวกรรมศาสตร์มหาบัณฑิต ที่จุฬาลงกรณ์มหาวิทยาลัย เมื่อ พ.ศ. 2539

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย