สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

เครื่องเจาะรูควบคุมด้วยคอมพิวเตอร์

Computer - controlled Punching Machine

ธนัช พนมตั้ง

MR.THANUT PANOMTUNG

ภาคภูมิ แช่มวงษ์

MR.PACKPOOM CHAMWONG

ภาณุภัสร์ ตันศิริ

MR.PANUPAT TANSIRI

เลขหมู่.....
เลขทะเบียน.....36837
วัน, เดือน, ปี..29..ส.ค..2543

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต

สาขาวิศวกรรมเครื่องกล

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2542

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปีการศึกษา2542

เครื่องเจาะรูควบคุมด้วยคอมพิวเตอร์

โดย

นายธนัช พนมตั้ง

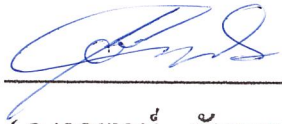
นายภาคภูมิ แช่มวงษ์

นายภาณุภัสร์ ตันศิริ

อาจารย์ที่ปรึกษา

(ผศ.พรศักดิ์ อรรถวานิช)

(อ.สกนธ์ คล่องบุญจิต)



(อ.เอกพจน์ ตันตราภิวัฒน์)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	เครื่องเจาะรูควบคุมด้วยคอมพิวเตอร์
นักศึกษา	ธนัช พนมตั้ง
	ภาคภูมิ แช่มวงษ์
	ภาณุภัสร์ ตันศิริ
ระดับการศึกษา	วิศวกรรมศาสตร์บัณฑิต สาขาวิศวกรรมเครื่องกล
	สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
พ.ศ.	2542
อาจารย์ผู้ควบคุมวิทยานิพนธ์	ผศ. พรศักดิ์ อรรถวานิช
	อ.สกนธ์ คล่องบุญจิต
	อ.เอกพจน์ ตันตราภิวัฒน์

บทคัดย่อ

เนื่องด้วยวิวัฒนาการในโลกปัจจุบันนี้ได้มีการพัฒนาคอมพิวเตอร์ให้มีความสามารถสูงขึ้น ทำให้คอมพิวเตอร์เข้ามามีบทบาทในชีวิตประจำวันเพิ่มมากขึ้นทดแทนและช่วยเหลือแรงงานจากมนุษย์และเครื่องจักรกลในด้านต่างๆ อาทิเช่น การคำนวณ การออกแบบ การควบคุม การตรวจสอบ การทดสอบ การแพทย์ เป็นต้น โดยเฉพาะในเรื่องการควบคุมนั้น เราได้มีการนำเอาคอมพิวเตอร์มาควบคุมเครื่องจักรกลให้ทำงานโดยอัตโนมัติแทนการควบคุมจากแรงงานคน ทั้งนี้เพื่อความถูกต้อง แม่นยำ ความละเอียดและความรวดเร็ว ดังนั้นโครงงานนี้จึงได้นำเอาความรู้ทางด้านคอมพิวเตอร์มาประยุกต์ใช้ร่วมกับความรู้ทางด้านกลไกต่างๆ โดยมีการเชื่อมต่อเข้าด้วยกันอย่างเหมาะสม ทั้งนี้เพียงแค่กำหนดค่าต่างๆลงบนหน้าจอคอมพิวเตอร์จากนั้น คอมพิวเตอร์จะประมวลผลตามโปรแกรมแสดงออกมาเป็นการทำงานของเครื่องเจาะแผ่นโลหะในส่วนต่างๆอย่างถูกต้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Thesis Title	Computer - controlled Punching Machine	
Student	Tanut	Panomtung
	Packpoom	Chamwong
	Panupat	Tansiri
Level of Study	Bachelor of Engineering in Mechanical Engineering	
	King Mongkut 's Institute of Technology Ladkrabang	
Year	1999	
Thesis Advisor	Pornsuk	Auttavanich
	Sakon	Klongboonjit
	Aekkapod	tuntrapiwat

ABSTRACT

Nowadays, computer has developed to have more ability, so computer has played an essential role in the different kinds of important aspects, replacing and helping human and machine in many different tasks such as Calculation, Designing, Control, Examination, Testing, Medical science and so on. Especially, on controlling machine, computer is used to automatically control the machine instead of human control because it gives more accuracy. Therefore, the highest aim of this study is focused on the combination of computer and mechanism knowledge. The results will be displayed on computer screen which the user can assign parameters for controlling machine. The computer will process the program and precisely display the operations of the Punching machine in every part.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สำเร็จลุล่วงไปได้เป็นอย่างดี เนื่องด้วยได้รับความอนุเคราะห์และให้คำแนะนำในหลายๆด้านจาก ผศ. พรศักดิ์ อรรถวานิช ดร.สรรพสิทธิ์ ลิ่มนรรัตน์ อ.เอกพจน์ ตันตราภิวัฒน์ อ.อุดม จันทร์จรัสสุข และ อ.พลชัย โชติปรายนกุล

ขอขอบคุณ คุณมณฑา เทียนเมือง และ คุณนิรุตต์ ร่วมกระโทก เป็นอย่างสูงที่ให้ความช่วยเหลือในด้าน Hardware ต่างๆ

ขอขอบคุณ คุณภาณุวัฒน์ เกรียงไกรวุฒิ, คุณสกล สมใจ และคุณป้อม ภาควิชาวิศวกรรมอิเล็กทรอนิกส์ ที่ให้การสนับสนุนในด้านการออกแบบวงจรต่างๆ

ขอขอบคุณ คุณพฤษถ์ ประเสริฐธรรม ภาควิชาวิศวกรรมไฟฟ้ากำลังที่ช่วยให้คำชี้แนะในด้านการเขียนโปรแกรมควบคุม

และต้องขอขอบคุณอีกหลายๆท่านที่ไม่ได้กล่าวถึง ณ ที่นี้ ที่ได้ให้คำแนะนำปรึกษาในด้านต่างๆเป็นอย่างดีมาโดยตลอด ทั้งนี้ต้องขอขอบคุณภาควิชาวิศวกรรมเครื่องกลเป็นอย่างสูงที่ได้ให้ทุนสนับสนุนการทำวิทยานิพนธ์ในครั้งนี้

นาย ธนัช พนมตั้ง
นาย ภาคภูมิ แช่มวงษ์
นาย ภาณุภัสร์ ตันศิริ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
บทที่ 1 บทนำ	1
บทที่ 2 ทฤษฎีเบื้องต้นเกี่ยวกับเครื่องจักรกลซีเอ็นซี	2
บทที่ 3 ทฤษฎีการอบชุบขั้นพื้นฐาน	11
บทที่ 4 ทฤษฎีเกี่ยวกับสเต็ปปิ้งมอเตอร์	14
บทที่ 5 ทฤษฎีเกี่ยวกับ Port Printer	28
บทที่ 6 อุปกรณ์ต่างๆ และ วัสดุในการทดลอง	36
บทที่ 7 โปรแกรมคอมพิวเตอร์	41
บทที่ 8 ขั้นตอนและผลการทดลอง	47
บทที่ 9 สรุปและแนวทาง	49
เอกสารอ้างอิง	
ภาคผนวก	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

เนื่องด้วยในปัจจุบันนี้จะเห็นได้ว่าคอมพิวเตอร์ได้เข้ามามีบทบาทในชีวิตประจำวันของคนเรามากขึ้นในหลายๆด้าน ไม่ว่าจะเป็นทางด้านการแพทย์ การสาธารณสุข การติดต่อสื่อสาร การคมนาคม การก่อสร้าง หรือแม้กระทั่งทางด้านอุตสาหกรรม ซึ่งในปัจจุบันได้มีอัตราการแข่งขันค่อนข้างสูง จึงเห็นได้ว่าแต่ละหน่วยงานก็ต่างมุ่งที่จะพัฒนาเทคโนโลยีทางด้านนี้กันอย่างเต็มที่ โดยเฉพาะทางด้านอุตสาหกรรม ได้มีการนำเอาคอมพิวเตอร์เข้ามามีบทบาทในการปฏิบัติงานเพิ่มมากขึ้น ทั้งทางด้านการออกแบบ การผลิต การบำรุงรักษา การตรวจสอบคุณภาพ หรือแม้กระทั่งการขนถ่ายผลผลิตไปยังผู้บริโภค คอมพิวเตอร์จึงเป็นกลจักรสำคัญที่จะทำให้หน่วยงานประสบความสำเร็จได้เป็นอย่างดี

การนำเอาคอมพิวเตอร์เข้ามาประยุกต์ใช้งานทางด้านการผลิตสามารถกระทำได้ในหลายๆส่วนด้วยกัน ซึ่งโครงงานนี้ก็เป็นตัวอย่างหนึ่งในการนำเอาคอมพิวเตอร์เข้ามาช่วยทางด้านการผลิต โดยการใช้คอมพิวเตอร์ควบคุมการทำงานของเครื่องเจาะโลหะแผ่น ซึ่งจะสามารถทำการเจาะโลหะแผ่นเป็นรูปต่างๆ เพื่อการนำไปใช้งานในภาคอุตสาหกรรมต่อไป อาทิเช่น งานขึ้นรูปโลหะแผ่นเป็นอุปกรณ์ต่างๆ เช่น ของเด็กเล่น เครื่องครัว ชิ้นส่วนรถยนต์ เครื่องมือแพทย์ เป็นต้น

ทั้งนี้เครื่องเจาะโลหะแผ่นหรือเครื่อง Punching เป็นเครื่องมือที่ใช้ในการเจาะแผ่นโลหะที่มีความหนาไม่มากเกินไป โดยทำการกำหนดตำแหน่งที่ต้องการลงบนหน้าจอคอมพิวเตอร์ จากนั้นคอมพิวเตอร์จะนำไปประมวลผลตามโปรแกรม ส่งสัญญาณควบคุมไปยังวงจรควบคุม Stepping Motor และหัวเจาะ ให้ทำงานตามขั้นตอนต่างๆจนเสร็จสมบูรณ์ ทั้งนี้ความถูกต้องแม่นยำจะขึ้นอยู่กับการวาง alignment ของ guide vane การจับยึดชิ้นส่วนต่างๆของเครื่องเจาะ และความละเอียดที่ได้จะขึ้นอยู่กับจำนวนสเต็ปของมอเตอร์ ความละเอียดของบอล์ลสกรู อย่างไรก็ดี ประสิทธิภาพในการเจาะจะใช้ได้ดีเมื่อชิ้นงานมีลักษณะเป็นแผ่นโลหะแข็งที่มีความหนาไม่มาก และมีขนาด(กว้าง - ยาว) พอเหมาะกับขนาดของที่จับยึดชิ้นงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีเบื้องต้นเกี่ยวกับเครื่องจักรกลซีเอ็นซี

2.1 ความหมายของเอ็นซีและซีเอ็นซี

เอ็นซี (NC) ย่อมาจากคำว่า Numerical Control หมายถึง การควบคุมเครื่องจักรกลด้วยระบบตัวเลขและตัวอักษร กล่าวคือ การเคลื่อนที่ต่างๆ ตลอดจนการทำงานอื่นๆของเครื่องจักรกลจะถูกควบคุมโดยรหัสคำสั่งที่ประกอบด้วยตัวเลข ตัวอักษร และสัญลักษณ์อื่นๆซึ่งจะถูกแปลงเป็นคลื่นสัญญาณ (pulse) ของกระแสไฟฟ้าหรือสัญญาณออกอื่นๆที่จะไปกระตุ้นมอเตอร์หรืออุปกรณ์อื่นๆ เพื่อทำให้เครื่องจักรทำงานตามขั้นตอนที่ต้องการ

ซีเอ็นซี (CNC) ย่อมาจากคำว่า Computerized Numerical Control ระบบควบคุมเอ็นซีแบบนี้จะมีคอมพิวเตอร์ที่มีความสามารถสูงเพิ่มเข้าไปภายในระบบทำให้สามารถจัดการกับข้อมูลที่ป้อนเข้าไปในระบบเอ็นซี และประมวลผลเพื่อนำผลลัพธ์ที่ได้ไปควบคุมการทำงานของเครื่องจักรกล

2.2 ความแตกต่างระหว่างเครื่องจักรกลเอ็นซีกับเครื่องจักรกลทั่วไป

ความแตกต่างในการใช้เครื่องจักรกลเอ็นซี เมื่อเปรียบเทียบกับเครื่องจักรกลที่ใช่ทั่วไปก็คือการตัดสินใจในการกำหนดขั้นตอนการทำงานต่างๆจะกระทำเพียงครั้งเดียว กล่าวคือ จะกระทำในขั้นตอนการวางแผนและสร้างโปรแกรมสำหรับควบคุมเครื่องจักรกลเท่านั้น ต่อจากนั้น โปรแกรมก็จะถูกนำไปใช้ในการควบคุมการทำงานของเครื่องจักรกล สำหรับการผลิตชิ้นงานที่ต้องการ โดยสามารถทำการผลิตซ้ำๆกันกี่ครั้งก็ได้ตามต้องการ

			เครื่องจักรกลทั่วไป	เครื่องจักรกลซีเอ็นซี
1	การป้อนโปรแกรม	ขั้นเตรียมงาน	ไม่มี	มี
2	การจับยึดชิ้นงาน	ขั้นเตรียมงาน	มือ	มือ
3	การจับยึดเครื่องมือ	ขั้นเตรียมงาน	มือ	มือหรือชุดควบคุม
4	การตั้งจุดอ้างอิง	ขั้นเตรียมงาน	มือ	มือ
5	การตั้งความเร็วรอบ	ขั้นเตรียมงาน	มือ	ระบบควบคุม
6	การเลื่อนแท่นเลื่อน	ขั้นทำงาน	มือหมุน	ระบบควบคุม
7	การเปรียบเทียบระยะ	ขั้นทำงาน	สายตา	ระบบควบคุม
8	การตรวจสอบขนาด	ขั้นทำงาน	เครื่องมือวัด	ใช้เวลาน้อยกว่า

ตาราง 2.1 ตารางเปรียบเทียบระหว่างระบบเอ็นซีกับระบบซีเอ็นซี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 ความแตกต่างระหว่างระบบเอ็นซีกับระบบซีเอ็นซี

ระบบซีเอ็นซีเป็นระบบที่พัฒนาต่อเนื่องมาจากระบบเอ็นซี ดังนั้น ความแตกต่างระหว่างระบบเอ็นซี ก็จะอยู่ที่ความสามารถของระบบควบคุม นั่นคือคอมพิวเตอร์เมื่อนำระบบซีเอ็นซีไปควบคุมเครื่องจักรกล ความสามารถในการทำงานต่างๆ จะเพิ่มมากขึ้นเมื่อเปรียบเทียบกับเครื่องจักรกลเอ็นซีดังนี้

1. การแสดงภาพจำลอง (Simulation) การทำงานตามโปรแกรมที่ป้อนเข้าในระบบทางจอภาพ
2. ความจุของหน่วยความจำเพิ่มมากขึ้น สามารถเก็บข้อมูลโปรแกรมได้มาก
3. การแก้ไขและลบโปรแกรมสามารถกระทำได้ที่เครื่องจักรโดยตรง
4. สามารถส่งข้อมูลไปเก็บไว้ในหน่วยความจำภายนอกได้
5. ระบบความปลอดภัยเพิ่มมากขึ้น
6. มีการชดเชยความผิดพลาดที่เกิดจากการวัดและการส่งกำลัง
7. มีโปรแกรมสำเร็จสำหรับการคำนวณค่าต่างๆเช่น ความเร็วรอบ อัตราป้อน เป็นต้น

2.4 ข้อดีและข้อเสียของเครื่องจักรกลเอ็นซีและซีเอ็นซี

ข้อดี ของเครื่องจักรกลเอ็นซีและซีเอ็นซี เมื่อเปรียบเทียบกับเครื่องจักรกลอัตโนมัติประเภทอื่นๆพอจะสรุปได้ดังนี้

1. มีความยืดหยุ่นในการทำงานสูง การเปลี่ยนงานใหม่จะแก้ไขหรือเปลี่ยนแปลงเฉพาะโปรแกรมเท่านั้น
2. ความเที่ยงตรง (Accuracy) จะอยู่ระดับเดียวกันตลอดช่วงความเร็วรอบและอัตราป้อนที่ใช้ทำการผลิต
3. เวลาในการผลิต (Production Time) สั้นกว่า
4. สามารถใช้ผลิตชิ้นงานที่มีรูปทรงซับซ้อนได้ง่าย
5. การปรับตั้งเครื่องจักรกระทำได้ง่าย ใช้เวลาน้อยกว่าการผลิตด้วยวิธีอื่นๆ
6. หลีกเลี่ยงความจำเป็นที่ต้องใช้ช่างควบคุมที่มีทักษะและประสบการณ์สูง
7. ช่างควบคุมเครื่องมเวลาว่างจากการควบคุมเครื่อง สามารถที่จะจัดเตรียมงานอื่นๆไว้ล่วงหน้าได้
8. การตรวจสอบคุณภาพไม่จำเป็นต้องกระทำทุกขั้นตอนและทุกชิ้น

ส่วน ข้อเสีย ของเครื่องจักรกลเอ็นซีและซีเอ็นซีมีดังนี้

1. ราคาของเครื่องจักรกลค่อนข้างสูง
2. การบำรุงรักษามีความซับซ้อนมาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. จำเป็นต้องใช้ช่างเขียนโปรแกรม (Part Programmer) ที่มีทักษะสูงและฝึกอบรมมาโดยเฉพาะ

4. ชิ้นส่วนหรืออะไหล่ที่ใช้ในการซ่อมบำรุงไม่สามารถผลิตได้ในประเทศ จำเป็นต้องสั่งซื้อหรือนำเข้าจากต่างประเทศ

5. การซ่อมบำรุงจะต้องใช้ช่างที่มีประสบการณ์สูงและผ่านการฝึกอบรมมาโดยเฉพาะ

6. ราคาของเครื่องมือต่างๆที่ใช้ในกระบวนการตัดเฉือน เช่น แกนเพลายึดมีดกัด มีดกลึงแบบใช้อินเสิร์ท (Insert) เป็นต้น มีราคาสูง

7. พื้นที่ติดตั้งเครื่องจักรจะต้องควบคุมระดับอุณหภูมิ ความชื้น และฝุ่นละออง

ข้อมูลเหล่านี้ จำเป็นจะต้องพิจารณาอย่างรอบคอบก่อนที่จะพิจารณาจัดซื้อ ซึ่งสามารถสอบถามได้จากบริษัทผู้ผลิตหรือตัวแทนจำหน่ายได้โดยตรง

2.5 การทำงานของเครื่องจักรกลเอ็นซี

หลักการทำงานของเครื่องจักรกลเอ็นซีหรือซีเอ็นซี จะคล้ายคลึงกับเครื่องจักรกลทั่วไป กล่าวคือโดยพื้นฐานเบื้องต้นแล้วเครื่องจักรกลเอ็นซีก็จะทำการผลิตชิ้นงานเหมือนกับเครื่องจักรกลทั่วไป เช่นเครื่องกัดเอ็นซี ก็จะทำงานเหมือนกับเครื่องกัดทั่วไป เพียงแต่ว่าระบบควบคุมเอ็นซีของเครื่องจะทำงานในขั้นตอนต่างๆแทนช่างควบคุมเครื่อง อย่างไรก็ตาม ก่อนที่เครื่องจักรกลเอ็นซีจะสามารถทำงานได้นั้น ระบบควบคุมของเครื่องจะต้องได้รับการบอกกล่าวเสียก่อนว่าจะให้ทำอะไร และจะต้องบอกกล่าวเป็นภาษาที่ระบบควบคุมสามารถเข้าใจได้ นั่นคือ จะต้องป้อนโปรแกรมเข้าไปในระบบควบคุมของเครื่องผ่านแป้นพิมพ์ (keyboard) หรือเทปแม่เหล็ก (magnetictape) ก็ได้

เมื่อระบบควบคุมอ่านโปรแกรมที่ป้อนเข้าไปแล้ว ก็จะนำไปควบคุมให้เครื่องจักรกลทำงาน แต่เนื่องจากเครื่องจักรกลเอ็นซีไม่มีมือสำหรับหมุนมือหมุนให้แท่นเลื่อนเคลื่อนที่ได้ ดังนั้นแท่นเลื่อนต่างๆจะต้องมีมอเตอร์ป้อน (feed motor) ประกอบอยู่ เช่น เครื่องกัดซีเอ็นซีจะมีการเคลื่อนที่ 3 แนวแกน ก็จะมีมอเตอร์ป้อน 3 ตัว

เมื่อระบบควบคุมอ่านโปรแกรมแล้ว ก็จะเปลี่ยนรหัสโปรแกรมนั้นให้เป็นสัญญาณทางไฟฟ้า เพื่อไปควบคุมให้มอเตอร์ทำงาน แต่เนื่องจากสัญญาณที่ออกจากระบบควบคุมนี้มีกำลังน้อย ไม่สามารถไปหมุนขับให้มอเตอร์ทำงานได้ ดังนั้น จึงต้องส่งสัญญาณนี้เข้าไปในภาคขยายสัญญาณของระบบขับ (drive amplified) และส่งต่อไปยังมอเตอร์ป้อนของแนวแกนที่ต้องการเคลื่อนที่

ความเร็วและระยะทางการเคลื่อนที่ของแท่นเลื่อน จะต้องกำหนดให้ระบบควบคุมรู้ ช่างควบคุมเครื่องอาศัยสายตามองดูตำแหน่งของคมตัดกับชิ้นงาน ก็จะรู้ว่าจะต้องเลื่อนแท่นเลื่อนไปอีกเป็นระยะทางเท่าใด แต่ระบบควบคุมเอ็นซีมองไม่ได้ ดังนั้น จึงต้องออกแบบอุปกรณ์หรือเครื่องมือที่สามารถจะบอกตำแหน่งของแท่นเลื่อนให้ระบบควบคุมรู้ได้ อุปกรณ์ชุดนี้เรียกว่า ระบบวัดขนาด (Measuring System) ซึ่งประกอบด้วยสเกลแนวตรง (Linear Scale) มีจำนวนเท่ากับจำนวนแนวแกน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการเคลื่อนที่ของเครื่องจักรกล ทำหน้าที่ส่งสัญญาณไฟฟ้าที่สัมพันธ์กับระยะทางที่แท่นเลื่อนเคลื่อนที่กลับไปยังระบบควบคุม ทำให้ระบบควบคุมรู้ว่าแท่นเลื่อนเคลื่อนที่ไปเป็นระยะทางเท่าใดแล้ว

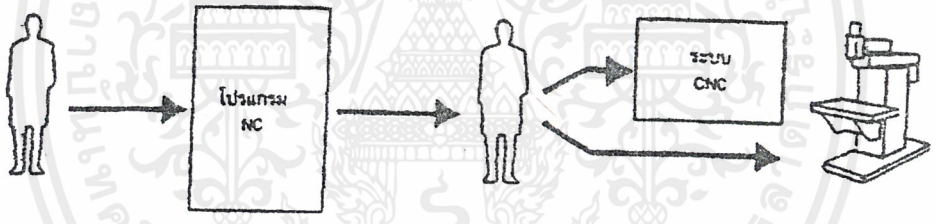
จากหลักการควบคุมการทำงานดังกล่าว ทำให้เครื่องจักรกลซีเอ็นซีสามารถผลิตชิ้นงานให้มีรูปทรงและขนาดที่ต้องการได้ จากลักษณะสร้างและการทำงานที่เหนือกว่าเครื่องจักรกลทั่วไป ทำให้เครื่องกลเอ็นซีและซีเอ็นซีเป็นปัจจัยหนึ่งที่มีความสำคัญมากในอุตสาหกรรมอัตโนมัติ และมีปริมาณความต้องการใช้เพิ่มมากขึ้นเรื่อยๆ

2.6 ระบบควบคุมเครื่องจักรกลด้วยตัวเลข

เครื่องจักรกลซีเอ็นซี จะประกอบด้วยองค์ประกอบใหญ่ๆอยู่ 2 ส่วน คือ

1. เครื่องจักรกล เป็นส่วนที่ทำหน้าที่ตัดเฉือนเจาะชิ้นงาน ตามขั้นตอนการทำงานที่กำหนดไว้

2. ระบบซีเอ็นซี เป็นส่วนที่ทำหน้าที่ควบคุมขั้นตอนการตัดเฉือนเจาะทั้งหมด



รูปที่ 2.1 องค์ประกอบของเครื่องจักรกล CNC

ข้อมูลเดิมที่อธิบายรายละเอียดของขั้นตอนที่ใช้ในการตัดเฉือนเจาะชิ้นงาน จะถูกป้อนเข้าไปในระบบควบคุมของเครื่องจักรกลก่อน ในรูปแบบของโปรแกรมเอ็นซี ซึ่งถูกจัดเตรียมโดยช่างเขียนโปรแกรม ช่างควบคุมเครื่องจะเป็นผู้ป้อนโปรแกรมเข้าไปในระบบควบคุม ซึ่งอาจป้อนด้วยมือผ่านแป้นพิมพ์โดยตรง หรือใช้แถบกระดาษเจาะรู (punched tape) ก็ได้ หลังจากนั้น ก็จะเดินเครื่องทดลองโปรแกรม และสังเกตสภาวะการตัดเฉือนเจาะชิ้นงานในแต่ละขั้นตอน บ่อยครั้งที่ช่างควบคุมเครื่องจะต้องจัดเตรียมโปรแกรม หรือเขียนโปรแกรมด้วยตนเอง หรือแก้ไขปรับปรุงโปรแกรมให้มีประสิทธิภาพในการตัดเฉือนเจาะสูงสุด ดังนั้น จึงเป็นสิ่งจำเป็นที่ช่างควบคุมเครื่องจะต้องมีความรู้ทั้งระบบควบคุมของเครื่องจักรกลและการเขียนโปรแกรมเอ็นซีด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7 องค์ประกอบของเครื่องจักรกลที่ควบคุมได้

องค์ประกอบหรือชิ้นส่วนของเครื่องจักรกล ที่ทำหน้าที่เคลื่อนที่เข้าตัดเฉือนชิ้นงาน และองค์ประกอบอื่นๆ ที่ช่วยเสริมการทำงานตัดเฉือนให้สมบูรณ์ยิ่งขึ้น จะถูกควบคุมโดยโปรแกรมเอ็นซี ด้วยวิธีการควบคุมแบบต่างๆกัน

2.7.1แนวแกนป้อน (Feed axes)

ในการกล่าวถึงเครื่องจักรกลซีเอ็นซี บ่อยครั้งที่เราจะได้ยินคำว่า แนวแกน (axes) ซึ่งหมายถึง แนวแกนการเคลื่อนที่ขององค์ประกอบของเครื่องจักรกล เช่น โต๊ะงาน เพลาหัวเครื่อง อุปกรณ์ลำเลียงเครื่องมือ (Tool carriers) เป็นต้น

สำหรับเครื่องจักรกลทั่วไป การเคลื่อนที่ในแนวแกนต่างๆ จะเกิดจากการหมุนมือ หมุนคันโยกหรือโยกคันโยกอัติโนมัติ (Feed levers)

เครื่องจักรกลซีเอ็นซีมีแนวแกนป้อนรวมกันอยู่หลายแนวแกนทำให้สามารถตัดเฉือนเจาะชิ้นงานให้เป็นรูปทรงต่างๆที่ต้องการได้ การกำหนดแนวแกนต่างๆของเครื่องจักรกลซีเอ็นซีจะกำหนดตามมาตรฐานสากลภายใต้หัวเรื่อง Coordinate axes and direction of movement for numerically controlled machinery ซึ่งจะกำหนดแนวแกนเหล่านี้โดยใช้ตัวอักษร x,y และ z

สำหรับเครื่องจักรกลซีเอ็นซี ที่ใช้ผลิตชิ้นงานที่มีรูปทรงซับซ้อนมาก จะมีจำนวนแนวแกนป้อนเพิ่มมากขึ้น

2.7.2การขับป้อน (Feed drives)

การเคลื่อนที่เรียงลำดับกันหรือพร้อมๆกันอย่างต่อเนื่องของแนวแกนป้อน จะทำให้เกิดการตัดเฉือนเจาะของเครื่องมือในชิ้นงาน

การขับป้อนจะทำให้เกิดการเคลื่อนที่ของแท่นเลื่อน ในขณะตัดเฉือนเจาะแท่นเลื่อนอาจพาให้ชิ้นงานเคลื่อนที่หรือคมตัดเคลื่อนที่ก็ได้

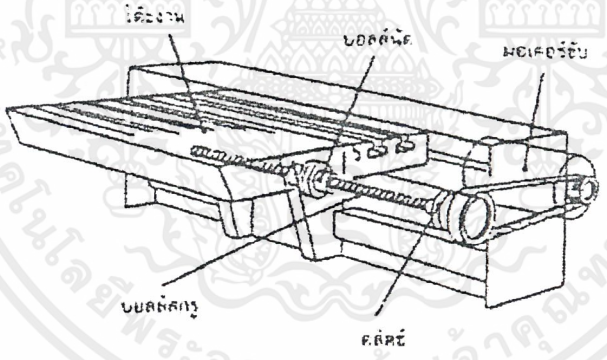
ระบบขับป้อนโดยทั่วไปจะใช้มอเตอร์กระแสตรงในการหมุนขับและควบคุมการทำงานด้วยวงจรอิเล็กทรอนิกส์จากภายนอก มอเตอร์ชนิดนี้จะสามารถหมุนและเบรคให้หยุดได้ทั้งสองทิศทางขณะตัดเฉือนเจาะชิ้นงาน การเคลื่อนที่ป้อนจะต้องเป็นไปอย่างสม่ำเสมอและสามารถต้านแรงกระทำจากภายนอกได้ เช่น แรงตัดเฉือน เป็นต้น ด้วยเหตุนี้ ระบบขับป้อนจึงต้องได้รับการออกแบบให้มีความแข็งแกร่งสูง มีการเคลื่อนที่คงที่และสม่ำเสมอ สามารถตอบสนองต่อการเปลี่ยนอัตราป้อนได้อย่างรวดเร็ว นอกจากนี้ในขณะทำงานคมตัดอาจทื่อ หรือการเคลื่อนที่ของแท่นเลื่อนถูกกีดขวาง หรือการเร่งอัตราป้อนให้เคลื่อนที่เร็วและหยุดโดยทันทีทันใด สาเหตุเหล่านี้จะทำให้มอเตอร์รับภาระมากเกินไป (over loading) ซึ่งอาจทำให้มอเตอร์เสียหายได้ ดังนั้น จึงต้องมีการป้องกันอุบัติเหตุเหล่านี้ โดยทั่วไปแล้วจะใช้คลัตท์แบบลูกกลิ้ง (Over running clutch) ร่วมกับ วงจรอิเล็กทรอนิกส์ ปัจจัยหนึ่งที่จะทำให้ระบบขับป้อนให้เหมาะสมกับการทำงานของเครื่องจักรและการออกแบบวงจรควบคุมการทำงานที่มีประสิทธิภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มอเตอร์แบบเป็นขั้น (Stepping motors) เป็นมอเตอร์ที่ทำงานแบบต่อเนื่อง โดยการแปลงคลื่นสัญญาณที่ป้อนเข้าไปในระบบให้เป็นการเคลื่อนที่เชิงมุม การหมุนในแต่ละมุมหรือขั้นที่เปลี่ยนไป 1 ขั้นจะเท่ากับ 1 คลื่นสัญญาณ ดังนั้น ตำแหน่งของเพลาจะถูกกำหนดโดยจำนวนคลื่นสัญญาณที่ป้อนเข้าไปในระบบ และความเร็วในการหมุนของเพลาจะวัดเป็นจำนวนขั้นต่อวินาที (Steps per second) ซึ่งจะเท่ากับความถี่ของคลื่น สัญญาณที่ป้อนเข้าในระบบที่วัดเป็นจำนวนคลื่นสัญญาณต่อวินาที (pulses per second) ความเที่ยงตรงของระบบจะขึ้นอยู่กับความสามารถของมอเตอร์ในการแบ่งขั้นการหมุนตามจำนวนคลื่นสัญญาณที่ป้อนเข้าในระบบแรงบิดของมอเตอร์ชนิดนี้จะลดลงเมื่อความเร็วในการหมุนแบ่งเพิ่มขึ้น ดังนั้น จึงเหมาะสำหรับเครื่องจักรกลเล็กๆที่ไม่ต้องใช้กำลังขับมาก เช่น เครื่องพลอตเตอร์ (Plotter machine) เป็นต้น

2.7.3บอลล์สกรู (Ball screws)

หัวใจของระบบขับป้อนของเครื่องจักรกลซีเอ็นซี ก็คือ การส่งกำลังขับด้วย บอลล์สกรู ซึ่งจะมีลูกบอลล์ไหลหมุนเวียนอยู่ตลอดเวลา บอลล์สกรูจะประกอบด้วยสกรูกับนัตที่มีลักษณะเป็นเกลียวกลม ร่องเกลียวกลมบนสกรูและในนัตจะชุบแข็งและเจียระไนผิวเรียบมันเพื่อลดความฝืดและเพิ่มความเที่ยงในการเคลื่อนที่ ดังแสดงในรูป

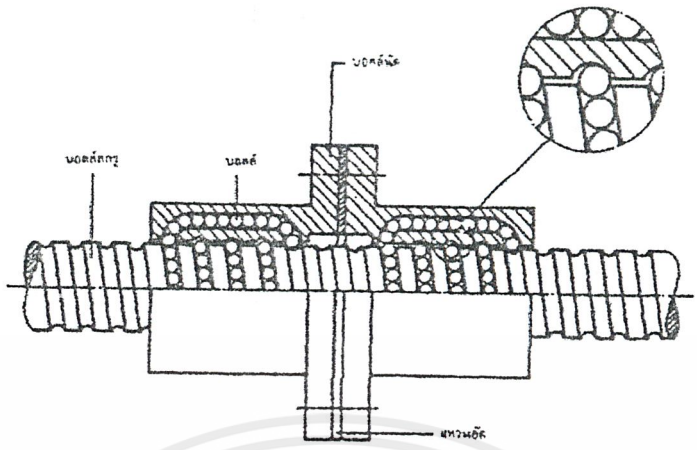


รูปที่ 2.2 การขับป้อนของโต๊ะงาน

เมื่อมอเตอร์หมุนขับสกรู นัตก็จะเคลื่อนที่ไปตลอดความยาวของสกรูพาให้แท่นเลื่อนและโต๊ะงานเคลื่อนที่ไปตามรางเลื่อน

ภายในของตัวนัตจะประกอบไปด้วยชุดของลูกบอลล์จำนวนมาก ดังแสดงในรูป ทำให้มั่นใจได้ว่าความเสียดทานในการส่งกำลังขับจากสกรูไปยังแท่นเลื่อนจะมีน้อยมากนัตจะถูกแบ่งออกเป็นสองซีก และ ขันประกอบยึดเข้าด้วยกัน โดยมีการเตรียมอัดแรงไว้ก่อน (preloaded) ทำให้สามารถลดระยะคลอน (backlash) ให้เหลือน้อยที่สุดจนแทบจะไม่มีเลยได้ทำให้การเคลื่อนที่ของแท่นเลื่อนมีความเที่ยงตรงสูงสามารถทำงานซ้ำๆกันได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.3 ลักษณะสร้างภายในของชุดบอลล์สกรู

การต่อกำลังระหว่างมอเตอร์กับบอลล์สกรู จะมีชุดคลัตช์ความฝืดเป็นตัวเชื่อม ซึ่งนอกจากจะมีหน้าที่ต่อกำลังขับแล้ว ยังมีหน้าที่ป้องกันอุบัติเหตุที่เกิดจากแท่นเลื่อนหรือโต๊ะงานชนหรือกระแทกกับสิ่งกีดขวางไม่ให้เครื่องจักรกลซีเอ็นซีเกิดความเสียหายมากเกินไป กล่าวคือ เมื่อมีการชน หรือกระแทกกันขึ้นจนแรงมากถึงค่าหนึ่ง ชุดคลัตช์ก็จะตัดระบบการส่งกำลังขับระหว่างมอเตอร์กับตัวบอลล์สกรูทันที

2.7.4 อุปกรณ์จับยึดชิ้นงาน (Workpiece holding devices)

อุปกรณ์ชิ้นงานจะจัดเตรียมไว้สำหรับยึดชิ้นงานเข้ากับโต๊ะงานในงานกัด (Milling) สามารถเลือกใช้อุปกรณ์จับยึดชิ้นงานแบบต่างๆกันได้ดังนี้

- แขนกดชิ้นงาน
- แองเกิล เพลท (angle plate)
- ปากกาจับชิ้นงาน
- แท่นแม่เหล็ก
- อุปกรณ์จับชิ้นงานที่ออกแบบเฉพาะงาน

2.8 ระบบควบคุมซีเอ็นซี (CNC Control System)

ระบบซีเอ็นซี (CNC system) จะมีคอมพิวเตอร์ประกอบอยู่ด้วยดังนั้น ช่างควบคุมเครื่องไม่เพียงแต่จะสามารถใช้โปรแกรมเอ็นซีสั่งให้เครื่องจักรทำงานได้เท่านั้น แต่จะยังสามารถเขียนและป้อนโปรแกรมด้วยตนเอง ตลอดจนการแก้ไขโปรแกรมได้หลังจากป้อนเข้าไปในระบบควบคุมของเครื่องแล้ว

ขนาดต่างๆของเครื่องมือตัดและอุปกรณ์จับยึดชิ้นงานสามารถที่จะเลือกใช้และป้อนเข้าไปในระบบควบคุมซีเอ็นซี ขณะทำการปรับตั้ง (setting-up) และเป็นอิสระจากตัวโปรแกรมเอ็นซี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขนาดต่างๆของเครื่องมือจะถูกนำไปใช้โดยอัตโนมัติในขณะทำการตัดเฉือนเจาะ ด้วยเหตุนี้ช่างควบคุมเครื่องจึงไม่จำเป็นต้องมีข้อมูลในการปรับตั้งมากและสามารถที่จะเลือกใช้เครื่องมือและอุปกรณ์จับยึดชิ้นงานได้ด้วยตนเอง

หากพิจารณาถึงภาษาโปรแกรม (Programming language) และเทคโนโลยีทางด้านการตัดเฉือนของเครื่องจักรกลที่ใช้ในระบบเอ็นซีกับซีเอ็นซีแล้วจะไม่แตกต่างกัน

2.8.1 ชนิดของการควบคุม (Control modes)

ลักษณะการควบคุมการเคลื่อนที่ทำงานของแท่นเลื่อนต่างๆในเครื่องจักรกล เอ็นซีและซีเอ็นซี จะมีการเคลื่อนที่อยู่ 2 ลักษณะ คือ

1. การเคลื่อนที่ในแนวเส้นตรง (Linear Interpolation หรือ straightline Interpolation) การเคลื่อนที่ลักษณะนี้ ระบบซีเอ็นซีจะคำนวณหาตำแหน่งของจุดต่างๆที่ต่อกันเป็นลูกโซ่ในแนวเส้นตรงระหว่างตำแหน่งของเครื่องมือ 2 ตำแหน่งในขณะที่เครื่องมือเคลื่อนที่จากจุดหนึ่งไปยังอีกจุดหนึ่งนั้น ระบบควบคุมซีเอ็นซีจะตรวจสอบและแก้ไขแนวแกนในการเคลื่อนที่ให้ถูกต้องอยู่ตลอดเวลา ทำให้การเคลื่อนที่ของเครื่องมือไม่ผิดพลาดหรือคลาดเคลื่อนออกจากจุดต่อของเส้นตรงมากกว่าค่าพิกัดความเผื่อของเครื่องที่กำหนดไว้

2. การเคลื่อนที่ในแนวเส้นโค้ง (Circular Interpolation) ระบบควบคุมซีเอ็นซี จะคำนวณหาตำแหน่งของจุดต่างๆ ที่ต่อกันเป็นเส้นโค้งตามขนาดรัศมีที่กำหนด ระหว่างตำแหน่งของเครื่องมือที่กำหนดไว้ 2 ตำแหน่ง ระบบควบคุมจะอาศัยจุดเหล่านี้ในการตรวจสอบและแก้ไขแนวการเคลื่อนที่ของเครื่องมือให้ถูกต้องและอยู่ภายใน พิกัดความเผื่อของเครื่องจักรกลที่กำหนด

ในระบบควบคุมซีเอ็นซีจะแบ่งการควบคุมการเคลื่อนที่ทั้งสองลักษณะตามลักษณะการเคลื่อนที่ป้อนออกเป็น 3 ชนิด คือ

1. การควบคุมจุดต่อจุด (Point to point control)

การควบคุมแบบนี้จะควบคุมการเคลื่อนที่ของเครื่องมือระหว่างจุดสองจุดที่โปรแกรมไว้ในลักษณะการเคลื่อนที่เร็ว (Rapid traverse) โดยที่เครื่องมือจะต้องไม่สัมผัสชิ้นงาน

แนวแกนในการเคลื่อนที่ขึ้นอยู่กับชนิดของระบบควบคุม กล่าวคือ มอเตอร์ขับของระบบป้อนอาจจะเริ่มทำงานหลายๆแนวแกนพร้อมกัน หรือทำงานทีละแนวแกนจนกว่าจะเคลื่อนที่ถึงตำแหน่งของเครื่องมือที่โปรแกรมไว้ ทำให้ไม่สามารถควบคุมทางเดินของเครื่องมือ (Tool path) การควบคุมแบบจุดต่อจุดมักจะใช้กับเครื่องเจาะ (Drilling machine) เครื่องเชื่อมจุด (Spot drilling) เป็นต้น

2. การควบคุมการตัดเฉือนแนวเส้นตรง (Straight – cut control)

การควบคุมชนิดนี้ นอกจากจะสามารถควบคุมการเคลื่อนที่ของเครื่องมือแบบเคลื่อนที่เร็วได้แล้ว ยังสามารถควบคุมการเคลื่อนที่ของเครื่องมือในแนวขนานกับแนวแกนของเครื่องจักรกลตามค่าอัตราป้อนที่ต้องการได้อีกด้วย แต่จะสามารถควบคุมการเคลื่อนที่ได้ครั้งละ 1 แนวแกนเท่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นั้น การเคลื่อนที่ของเครื่องมือจะถูกควบคุมด้วย อัตราป้อนและความยาวในการเคลื่อนที่ ระบบการควบคุมการตัดเฉือนแนวเส้นตรงชนิดนี้จะใช้กับเครื่องกัดและเครื่องกลึงแบบง่ายๆ

3. การควบคุมตามเส้นขอบรูป (Contouring control)

การควบคุมแบบนี้จสามารถควบคุมการเคลื่อนที่ทำงานได้ดังนี้

- ควบคุมเครื่องมือให้เคลื่อนที่ไปยังตำแหน่งที่ต้องการแบบเคลื่อนที่เร็วได้
- ควบคุมเครื่องมือให้เคลื่อนที่ขนานกับแนวแกนไปยังตำแหน่งที่ต้องการตามค่าอัตราป้อนได้
- ควบคุมเครื่องมือให้เคลื่อนที่ไปยังตำแหน่งใดๆบนชิ้นงานที่กำหนดในแนวแกนเส้นตรงและเส้นโค้งตามค่าอัตราป้อนได้

2.8.2 การควบคุมหน้าที่การทำงานของเครื่องจักรกล (Control of machine function)

ระบบควบคุมซีเอ็นซีนอกจากจะสามารถควบคุมการเคลื่อนที่ของเครื่องมือตามรูปทรงเรขาคณิตของชิ้นงานแล้ว ยังสามารถควบคุมหน้าที่การทำงานอื่นๆที่ช่วยเสริมการทำงานตัดเฉือนของเครื่องจักรกลให้เหมาะสมกับสภาวะการทำงานในขณะนั้นได้อีกด้วย จำนวนหน้าที่การทำงานและวิธีการควบคุมจะไม่ขึ้นอยู่กับตัวเครื่องจักรกลเพียงอย่างเดียว แต่ยังขึ้นอยู่กับชนิดระบบควบคุมอีกด้วย

ตัวอย่างหน้าที่การทำงานต่างๆ ที่จำเป็นจะต้องโปรแกรมเพื่อช่วยในการทำงานมีดังนี้

- การเริ่มหมุนของเพลางาน ทิศทางการหมุนและการเปลี่ยนความเร็วรอบ
- การกำหนดตำแหน่งของเพลางาน
- การเปิดสารหล่อเย็นและความดันของสารหล่อเย็น
- การรักษาอัตราป้อนให้คงที่
- การเปลี่ยนตำแหน่งของเครื่องมือ
- การรักษาความเร็วตัดให้คงที่
- การเริ่มทำงานหรือควบคุมการทำงานของอุปกรณ์ช่วยงานอื่นๆ เช่น อุปกรณ์เปลี่ยนชิ้นงาน ได้แก่ โต๊ะเปลี่ยนงาน (pallet shuttle) เป็นต้น
- ชุดยันศูนย์ท้ายแท่น (Tail – stock)
- อุปกรณ์ใส่และถอดชิ้นงาน (loader and unloader)
- แท่นประคองศูนย์ (Steady rest)
- อุปกรณ์ลำเลียงเศษ (chip conveyor)
- Sorter

เครื่องจักรกลซีเอ็นซีที่สามารถใช้ระบบควบคุมสั่งการทำงานในหน้าที่ต่างๆได้ยิ่งมากเท่าใดก็จะเป็นระบบที่มีความเหมาะสมสำหรับการควบคุมแบบอัตโนมัติมากยิ่งขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
ทฤษฎีการอบชุบเหล็กกล้าขั้นพื้นฐาน
(Heat treatment of steels)

การอบชุบเหล็กในที่นี้มิได้หมายถึงเฉพาะการชุบเหล็กให้แข็งเพียงอย่างเดียว แต่หมายถึงการใช้ความร้อนกระทำต่อเหล็ก เพื่อเปลี่ยนแปลงคุณสมบัติของเหล็กและโครงสร้างภายในของเหล็ก การอบชุบเหล็กที่สำคัญและใช้กันมากมีดังนี้

3.1 การชุบแข็ง(Hardening)

การชุบแข็ง คือการชุบเหล็กเพื่อให้เหล็กมีความแข็งขึ้น ขั้นแรกจะต้องเผาเหล็กจากอุณหภูมิบรรยากาศให้ร้อนประมาณ 800 - 900 องศาเซลเซียส (ขึ้นอยู่กับชนิดของเหล็ก) ภายในเตาไฟฟ้าหรือเตาอบชุบอื่นๆที่สามารถควบคุมอุณหภูมิได้ เมื่อเหล็กร้อนจนถึงอุณหภูมินี้โครงสร้างของเหล็กจะเปลี่ยนไปเป็นออสเตนไน (ดูจาก diagram) ทั่วๆไป มักจะเผาให้อุณหภูมิของเหล็กร้อนเลยเส้น A3 ไปประมาณ 50 - 75 องศาเซลเซียส แต่จะไม่เผาเหล็กให้อุณหภูมิสูงเกินพิกัดนี้ เพราะจะทำให้เกรนของเหล็กขยายตัวโตขึ้น ซึ่งจะทำให้คุณภาพของเหล็กหลังจากการชุบเสียไป ในทางปฏิบัติถ้าเป็นเหล็กที่มีคาร์บอนต่ำกว่า 0.8% จะเผาให้อุณหภูมิสูงกว่าเส้น A3 ประมาณ 50 - 75 องศาเซลเซียส ถ้าเหล็กมีคาร์บอนสูงกว่า 0.8% จะเผาเพียงแค่อุณหภูมิเหนือเส้น A1 ประมาณ 50 - 75 องศาเซลเซียสเท่านั้น เหตุผลก็คือเพื่อต้องการรักษาซีเมนไตต์ไว้ไม่ให้เปลี่ยนไปเป็นออสเตนไนท์ เพราะซีเมนไทต์มีความแข็งสูงอยู่แล้ว (ดูภาพประกอบ)

เมื่อเผาเหล็กจนกลายเป็นออสเตนไนท์แล้วจะต้องทิ้งไว้ที่อุณหภูมินี้ระยะหนึ่ง เวลาประมาณ 1 ชั่วโมงต่อความหนา 25 มม. เพื่อให้เหล็กร้อนทั่วถึงกันตลอดทั้งแท่ง จากนั้นจึงนำเอาเหล็กออกจากเตาทำให้เย็นโดยรวดเร็วด้วยการจุ่มลงในน้ำ (Water quench) หรือในน้ำมัน (Oil quench) ในขณะที่ทำให้เหล็กเย็นโดยรวดเร็วนี้ ออสเตนไนท์จะเปลี่ยนกลับไปเป็นเฟอร์ไรท์และเพิร์ไลท์ไม่ทัน แต่จะให้โครงสร้างใหม่ที่มีความแข็งแรงสูง เรียกว่ามาร์เทนไซท์ ถ้าจะพิจารณาโดยละเอียดแล้วมาร์เทนไซท์ก็คือเหล็กเฟอร์ไรท์นั่นเอง แต่เป็นเหล็กเฟอร์ไรท์ที่มีปริมาณคาร์บอนสูงกว่าสภาพสมดุลย์ตาม Equilibrium diagram ที่เป็นเช่นนี้เพราะอะตอมของคาร์บอนไม่มีเวลาพอที่จะเคลื่อนไหวมารวมกันให้ซีเมนไตต์ ดังนั้นความแข็งของเหล็กที่ผ่านการชุบจึงขึ้นอยู่กับสองสิ่งคือ

1) ปริมาณธาตุคาร์บอนในเหล็ก กล่าวคือ ถ้าในเหล็กมีธาตุคาร์บอนมากโอกาสที่เหล็กจะเปลี่ยนเป็นมาร์เทนไซท์ก็ยิ่งง่าย และให้ปริมาณของมาร์เทนไซท์มาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

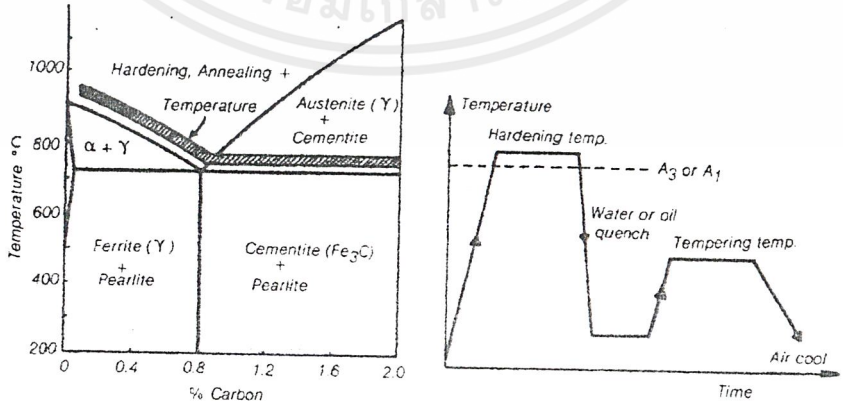
2) อัตราความเร็วในการชุบ กล่าวคือ ถ้าทำให้เหล็กเย็นเร็วๆ โอกาสที่ออสเตนไนท์จะเปลี่ยนเป็นมาร์เทนไซท์ก็มีมาก ในทางตรงกันข้ามถ้าปล่อยให้เหล็กเย็นช้าๆ ออสเตนไนท์จะเปลี่ยนไปเป็นเฟอร์ไรท์กับซีเมนไตต์หมด ไม่เกิดมาร์เทนไซท์ เหล็กก็ไม่แข็ง

3.2 การอบคืนไฟ (Tempering)

เหล็กที่ผ่านการชุบมาแล้วย่อมจะเกิดความเครียด (strain) ขึ้นภายใน และมีความแข็งเพิ่มขึ้น แต่เหล็กจะขาดคุณสมบัติทางด้านความเหนียว(Ductility) ทำให้ไม่เหมาะที่จะนำไปใช้งาน เพราะถ้าเกิดมีการกระแทกขึ้นเหล็กอาจจะแตกร้าวได้ จึงจำเป็นต้องปรับปรุงคุณสมบัติเสียใหม่โดยการทำอบไฟซึ่งมีวิธีการทำดังนี้

นำเหล็กที่ผ่านการชุบมาแล้วเผาภายในเตาที่อุณหภูมิต่ำประมาณ 200 - 400 องศาเซลเซียส ทิ้งไว้ประมาณ 1 - 3 ชั่วโมง แล้วเอาออกปล่อยให้เย็นในอากาศธรรมดา เหล็กจะมีคุณสมบัติทางด้านความเหนียว (Ductility) ดีขึ้น แต่ความแข็งจะลดลงเล็กน้อย ในขณะที่เผาที่อุณหภูมิต่ำ มาร์เทนไซท์จะมีการเปลี่ยนแปลงเล็กน้อย โดยอะตอมของคาร์บอนจะเคลื่อนไหวออกจากมาร์เทนไซท์มารวมกันเป็นเฟอร์ไรท์และซีเมนไตต์บางส่วน ที่เป็นเช่นนี้เพราะมาร์เทนไซท์ไม่ใช่โครงสร้างของเหล็กที่สมดุลย์ที่อุณหภูมิบรรยากาศ เมื่อเหล็กได้รับความร้อนจะเกิดการเปลี่ยนแปลงเพื่อกลับไปเป็นโครงสร้างที่สมดุลย์คือเหล็กเฟอร์ไรท์กับซีเมนไตต์ ที่เราต้องเผาที่อุณหภูมิไม่เกิน 400 องศาเซลเซียสก็เพื่อไม่ต้องการให้มาเทนไซท์คืนตัวหมด เพราะเรายังต้องการความแข็งของเหล็กอยู่ ถ้าเราเผาให้อุณหภูมิสูงเกิน 400 องศาเซลเซียส ความแข็งจะถูกทำลายหมด

โดยทั่วๆ ไปการชุบแข็งและการคืนตัวจะต้องกระทำติดต่อกัน เพื่อให้ได้เหล็กแข็งและทนแรงกระแทกได้ดีด้วย(ดูภาพประกอบ)



ภาพ 3.1 แสดงช่วงของอุณหภูมิในการทำการอบชุบแข็งและการอบคืนไฟ

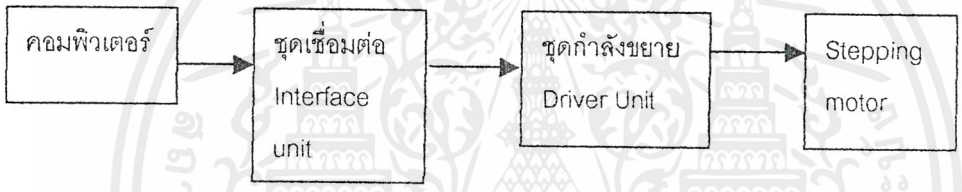
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทำหัว punch และ die ในโปรเจ็คนี้เราได้ใช้เหล็กชนิด HB(มาตรฐานเหล็กที่ใกล้เคียง คือ AIS = PATENTED, DIN = 1.7225, ASSAB = 709 ซึ่งมีคุณสมบัติใช้ทำแม่พิมพ์ยาง, Die, แคสติ้ง, ฐานรองแม่พิมพ์) มาทำการอบที่อุณหภูมิ 880 องศาเซลเซียส เป็นเวลา 30 นาที จากนั้นก็นำมาชุบแข็งในน้ำมัน ซึ่งมีค่าความแข็ง HRC ประมาณ 30-40

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
ทฤษฎีเกี่ยวกับสเต็ปปิ้งมอเตอร์

สเต็ปปิ้งมอเตอร์เป็นอุปกรณ์ จำพวกเชิงกลทางไฟฟ้าที่มีอินพุทเป็นกลุ่มของไบนารีโลเตจ และเป็นลักษณะของการเคลื่อนที่แบบเชิงมุม หรือหมุนไปตามสเต็ป (แต่ละสเต็ปอยู่ในช่วง 0.1-90 องศา ขึ้นอยู่กับโครงสร้างของสเต็ปปิ้งมอเตอร์) ตามสัญญาณพัลซ์ป้อนให้กับขั้วสเตเตอร์จึงจะเกิดแรงผลักต่อโรเตอร์ทำให้โรเตอร์หมุนไป แต่ลักษณะของสเต็ปปิ้งมอเตอร์จะมีขั้วของสเตเตอร์อยู่หลายขดซึ่งเรียกว่า " เฟส " นั้นเมื่อป้อนสัญญาณพัลซ์ในลักษณะซีเควนของเลขไบนารีโดยผ่านวงจรไดเวอร์ (Driver) ทำให้โรเตอร์หมุนได้อย่างต่อเนื่องกับบล็อคไดอะแกรมรูปที่ 4.1



รูปที่ 4.1 บล็อคไดอะแกรมแสดงการควบคุมสเต็ปปิ้งมอเตอร์

4.1 คุณประโยชน์ของสเต็ปปิ้งมอเตอร์

ในระบบควบคุมตำแหน่งใช้สเต็ปปิ้งมอเตอร์ มีข้อดีอยู่หลายประการคือ

-เป็นลักษณะการควบคุมแบบไม่ต้องการป้อนกลับไม่ว่าจะเป็นการควบคุมตำแหน่งหรือความเร็ว

-ความผิดพลาดเกี่ยวกับตำแหน่งแทบไม่มีเลยเนื่องจากการเคลื่อนที่ของสเต็ปปิ้งมอเตอร์นั้นเคลื่อนที่เป็นสเต็ปด้วยจำนวนองศาที่แน่นอน

-สเต็ปปิ้งมอเตอร์จะถูกนำมาใช้กับเครื่องมือที่ต้องการความละเอียดแม่นยำ และใช้อยู่ในเครื่องมือประเภทดิจิตอล เช่น เครื่องวาดรูป นิวเมอริคอคอนโทรล Computer numerical control (CNC)

-ไม่จำเป็นต้องใช้วงจรแปลงดิจิตอลเป็นอินเตอร์เฟสกับไมโครคอมพิวเตอร์

-สเต็ปปิ้งมอเตอร์เป็นอุปกรณ์ที่เปลี่ยนสัญญาณดิจิตอล ไปเป็นการเคลื่อนที่ทางกล ดังนั้นการติดต่อกับอุปกรณ์ดิจิตอลก็เป็นไปโดยง่าย และวงจรขยายกำลังจากสัญญาณดิจิตอล (digital power amplifier) ที่ใช้ก็ถูกกว่าวงจรขยายกำลังเชิงเส้น (linear power amplifier)อีกด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

-การออกแบบวงจรควบคุมสเต็ปปิ้งมอเตอร์สามารถทำได้ง่ายกว่าวงจรควบคุมมอเตอร์แแบเซอร์โว และยังสามารถออกแบบวงจรในสเต็ปปิ้งมอเตอร์ทำงานหรือหยุดได้แบบทันทีทันใด

4.2 คุณลักษณะของสเต็ปปิ้งมอเตอร์

ความถูกต้องเที่ยงตรงของมุมสเต็ปขณะที่ไม่มีโหลด จะถูกระบุสำหรับมอเตอร์ทุกชนิด เช่น มอเตอร์ที่มีสเต็ป 7.5 °ความผิพลาด ±10 ลิปดา ขระเคลื่อนที่ไปหนึ่งสเต็ป เป็นต้น

มุมสเต็ป	จำนวนสเต็ปต่อรอบ
0.9 °	400
1.8°	200
3.6°	100
3.75°	96
7.5°	48
15.0°	24

ตารางที่ 4.1 ตัวอย่างของมุมสเต็ป

มอเตอร์ที่มีจำนวนสเต็ปต่อรอบเท่ากับ 4 จะมีค่าผิดพลาดเป็นศูนย์ เมื่อหมุนครบ 1 รอบ เพราะขณะที่หมุนมา ณ ตำแหน่งเดิมขณะเริ่มต้นขั้วแม่เหล็กและทิศทางของเส้นแรงแม่เหล็ก (Flux) วงเดิมด้วยเหตุนี้การเปลี่ยนตำแหน่งของสเต็ปปิ้งมอเตอร์ที่ต้องการความถูกต้องสูงๆ จะต้องแบ่งจำนวนสเต็ปต่อรอบเป็นจำนวนเท่าของ สี่สเต็ป เพื่อลดการสะสมของค่าผิดพลาด (Step angle error) ซึ่งเป็นรูปแบบการทำงานแบบสี่สเต็ปตัวอย่างสเต็ปของมุมสเต็ปแสดงดังตารางที่ 4.1

แรงบิด (Torque)

การทำงานของสเต็ปปิ้งมอเตอร์มีแรงบิดเกี่ยวข้องอยู่3 ชนิดคือ

1. โฮลดิ้งทอร์ค(Holding Torque) คือแรงบิดที่ทำให้สเต็ปปิ้งมอเตอร์เริ่มหมุนไปสองสเต็ปขณะหยุดนิ่ง ถ้าแรงบิดที่ทำให้สเต็ปปิ้งมอเตอร์มีขนาดมากกว่าโฮลดิ้งทอร์ค จะทำให้มอเตอร์สูญเสียการหมุนแบบสเต็ปกลายเป็นการหมุนแบบต่อเนื่อง โดยปกติขณะทำงานของมอเตอร์จะน้อยกว่าระดับโฮลดิ้ง

2. ดิเทนซ์ทอร์ค (Detent Torque) ซึ่งเป็นสเต็ปปิ้งมอเตอร์แบบไฮบริดและแบบแม่เหล็กถาวร จะมีส่วนประกอบของโรเตอร์เป็นแม่เหล็กถาวรซึ่งจะสร้างแรงบิดมาเบรคการหมุนของมอเตอร์อย่างสม่ำเสมอในขณะที่ไม่มีการป้อนกระแสเข้าขดสเตเตอร์ แรงบิดดังกล่าวนี้เรียกว่า ดิเทนซ์ทอร์ค

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ไดนามิคทอร์ค (Dinamic or working torque) คือแรงบิดขณะทำงานซึ่งอาจเกิดการเปลี่ยนแปลงได้ เนื่องมาจากการปรับเปลี่ยนอัตราเร็วของมอเตอร์ โดยปกติการเปลี่ยนอัตราเร็วของมอเตอร์จะอยู่ในย่านระหว่าง เส้นโค้งพูลอิน (Pull-in curve) และเส้นโค้งพูลเอ้าท์ (Pull-out curve) เพราะถ้าปรับอัตราเร็ว ณ จุดนอกโค้งพูลเอ้าท์มอเตอร์จะสูญเสียการหมุนแบบเป็นสเต็ปได้หรือเกิดการหมุนแบบต่อเนื่องนั่นเอง

4.3 หลักการทำงานของสเต็ปปิ้งมอเตอร์

สเต็ปปิ้งมอเตอร์สามารถแบ่งโครงสร้างทางกายภาพออกได้เป็น 2 ส่วน คือ สเตเตอร์ (STATOR) และโรเตอร์(ROTOR) ตัวสเตเตอร์เป็นส่วนทที่อยู่กับที่ ประกอบด้วยขดลวดทองแดงซึ่งพันอยู่รอบแกนเหล็ก เพื่อสร้างสนามแม่เหล็กเมื่อมีการจ่ายกระแสผ่านขดลวด ส่วนโรเตอร์เป็นส่วนที่เคลื่อนที่ มีลักษณะเป็นแท่งเหล็กทรงกลม และที่ผิวรอบนอกมีลักษณะเป็นซีกฟันซึ่งทำจากแม่เหล็กถาวร

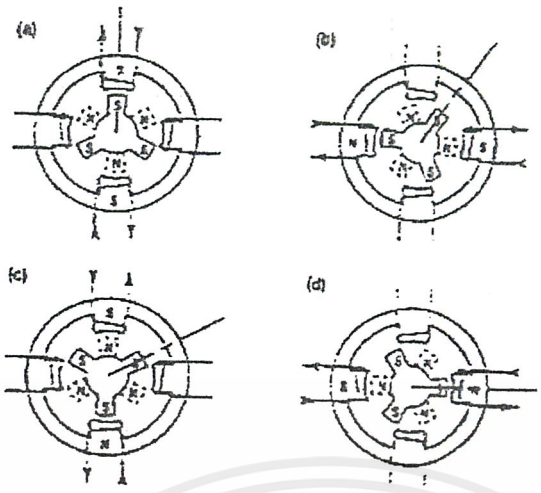
เมื่อยังไม่มีการจ่ายกระแสให้กับขดลวดของมอเตอร์ซีกฟันอันใดอันหนึ่งของโรเตอร์จะอยู่ในตำแหน่งที่ตรงกับซีกฟันอันใดอันหนึ่งของสเตเตอร์ ทั้งนี้เป็นเพราะแม่เหล็กถาวรที่วตัวของโรเตอร์พยายามที่จะทำให้ค่าความต้านทานทางแม่เหล็กไฟฟ้า (RELUCTANCE) มีค่าน้อยที่สุด ซึ่ง ณ ที่จุดที่ซีกฟันของตัวโรเตอร์และสเตเตอร์ตรงกันนั้นมีค่าความต้านทานทางแม่เหล็กไฟฟ้าน้อยที่สุด ทำให้เกิดเส้นแรงแม่เหล็กไฟฟ้ามากที่สุด เส้นแรงแม่เหล็กไฟฟ้าจะทำให้เกิดขั้วแม่เหล็กเหนือและใต้ขึ้นมา 2 คู่ ทั้งที่ตัวสเตเตอร์และตัวโรเตอร์ดังรูป ค่าทอร์ค (YORQUE) ที่ทำให้ตัวโรเตอร์ยึดอยู่ในตำแหน่งดังกล่าวนี้เรียกว่า ดีเท็นท์ทอร์ค (DETENT TORQUE) (หมายความว่า การที่จะทำให้มอเตอร์เคลื่อนที่ในขณะที่ไม่ได้จ่ายกระแสให้กับขดลวดของมอเตอร์จะต้องออกแรงมากกว่าค่าของดีเท็ฯท์ทอร์ค จึงทำให้โรเตอร์เคลื่อนที่ได้)

เมื่อจ่ายกระแสให้กับขดลวดที่อยู่ในสเตเตอร์คู่ใดคู่หนึ่ง ดังรูปที่ 4.2a จะทำให้เกิดขั้วแม่เหล็กเหนือและใต้ที่ซีกฟันของตัวสเตอร์ ซึ่งจะดึงดูดซีกฟันของตัวโรเตอร์ที่มีขั้วแม่เหล็กที่มีศักย์ต่างกันอยู่ใกล้ที่สุดเข้าไว้ ตำแหน่งนี้เรียกว่า สเตเบิ้ลโพสชิชั่น (STABLE POSITION) ของโรเตอร์จะมีจำนวนตำแหน่งเท่ากับซีกฟันของโรเตอร์ และแรงที่จะให้โรเตอร์เปลี่ยนตำแหน่งไปจากตำแหน่งสเตเบิ้ลโพสชิชั่นได้เรียกว่า โฮลดิ้งทอร์ค (HOLDING TORQUE)

เมื่อสับเปลี่ยนการจ่ายกระแสให้แก่ขดลวด จากขดหนึ่งไปยังอีกขดหนึ่ง เนื่องจากขดลวดวางอยู่ในตำแหน่งที่ต่างกัน 90 องศา ก็จะทำให้ตัวสเตเตอร์ดึงดูดซีกฟันของตัวโรเตอร์อีกซีกหนึ่งที่ใกล้ที่สุดเข้าไว้ ซึ่งจะทำให้ตัวโรเตอร์เคลื่อนที่ไป 1 สเต็ปหรือ 30 องศาดังรูปที่ 4.2b จากนั้นก็เปลี่ยนไปจ่ายกระแสให้กับขดลวดชุดแรกโดยในคราวนี้เปลี่ยนทิศทางการไหลของกระแสให้ตรงกันข้ามกับครั้งแรก ซึ่งจะทำให้ตัวโรเตอร์เคลื่อนที่ไปอีก 1 สเต็ป (เคลื่อนที่ไป 30 องศา)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง



รูปที่ 4.2 แสดงขั้นตอนการทำงานของสเต็ปปิ้งมอเตอร์แบบเต็มสเต็ปหนึ่งเฟสดังรูปที่ 4.2c หลังจากนั้นก็ไปจ่ายกระแสให้กับขดลวดชุดที่สองโดยกลับทิศทางของกระแสที่ป้อนให้อีกเช่นกัน ทำให้โรเตอร์หมุนไป 90 องศาดังรูปที่ 4.2d และถ้าหากเราป้อนกระแสให้กับมอเตอร์เหมือนที่เราป้อนในครั้งแรกแล้ว ซีกฟันซีกถัดไปของตัวโรเตอร์จะอยู่ในตำแหน่งที่เหมือนกับในรูปที่ 4.2a อีกครั้งหนึ่ง ถ้าหากเราต้องการเคลื่อนที่ 1 รอบ เราจะต้องกระตุ้นให้มอเตอร์เคลื่อนที่ไปจนครบ 12 สเต็ป และถ้าต้องการให้โรเตอร์หมุนไปอีกทิศทางหนึ่ง ก็จะทำการสลับลำดับในการจ่ายกระแสจากรูปที่ 4.2a,4.2d,4.2c,4.2b ตามลำดับ

4.4 โหมดการทำงานของ สเต็ปปิ้งมอเตอร์

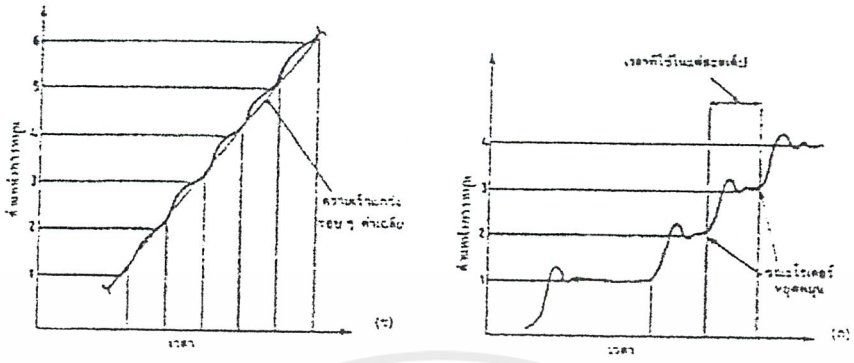
ถ้าจะแบ่งโหมดการทำงานของสเต็ปปิ้งมอเตอร์ ตามอัตราเร็วของสเต็ป จะแบ่งออกได้เป็น 2 โหมด คือ หมุนเป็นสเต็ป (Discrete Stepping Mode) และหมุนเป็นแบบต่อเนื่อง (Slewing Mode)

โดยถ้าการหมุนเป็นแบบสเต็ป และมีเวลาหยุดนิ่งก่อนที่จะเปลี่ยนเป็นสเต็ปต่อไป ก็จะเรียกการทำงานในโหมดนี้ว่าการหมุนเป็น สเต็ป ดังรูปที่ 4.3(ก) สำหรับตัวอย่างของเครื่องใช้ที่ทำงานในโหมดนี้คือ เครื่องเจาะบัตร การทำงานต่างๆก็คือ สเต็ปปิ้งมอเตอร์จะเป็นตัวส่งแถบกระดาษเข้าไปยังเครื่องปรุกระดาษเพื่อบันทึกข้อมูลลงในแทบกระดาษซึ่งการหมุนของสเต็ปปิ้งมอเตอร์ จะหมุนไปแล้วหยุดชั่วขณะเพื่อปรุกระดาษให้เรียบร้อยก่อน แล้วจึงคอยหมุนต่อไปยังตำแหน่งเจาะใหม่

ถ้าเพิ่มอัตราเร็วในแต่ละสเต็ปให้เร็วขึ้น และเป็นไปอย่างต่อเนื่องไม่มีการหยุดนิ่ง จะเรียกกำหนดการทำงานนี้ว่า การหมุนแบบต่อเนื่องดังแสดงในรูปที่ 4.3(ข) ซึ่งสามารถหาความสัมพันธ์ระหว่างความเร็วรอบมอเตอร์ (n) กับอัตราเร็วของสเต็ป (f) และจำนวนสเต็ปทั้งหมด (s) ได้ดังสมการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
36837

$$N = 60.(f/s) \qquad \text{.....(1)}$$



รูปที่ 4.3 กราฟแสดงสเต็ปการหมุนในโหมดการทำงาน

(ก) หมุนแบบสเต็ป

(ข) หมุนต่อเนื่อง

4.5 ชนิดของสเต็ปปิ้งมอเตอร์

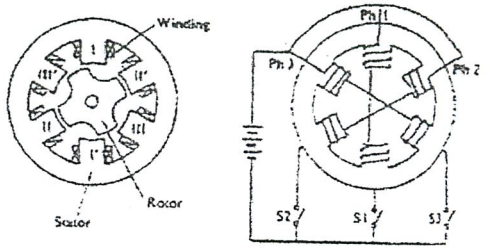
สเต็ปปิ้งมอเตอร์สามารถแบ่งออกได้หลายชนิดตามลักษณะของโครงสร้างและการใช้งาน ดังต่อไปนี้

4.5.1 สเต็ปปิ้งมอเตอร์ชนิดวาริเอเบิลรีลักแตนซ์(VARIABLE RELUCTANCE STEPPING MOTOR)

สเต็ปปิ้งมอเตอร์ชนิดนี้สามารถปรับค่ารีลักแตนซ์ได้ ซึ่งรูปที่ 4.4 แสดงภาพตัดขวางของสเต็ปปิ้งมอเตอร์แบบ 3 เฟส โดยที่สเตเตอร์มีฟันทั้งหมด 6 ซี่ซี่ที่อยู่ตรงข้ามกันหรือทำมุมกัน 180 องศา ซึ่งกันและกันจะเป็นเฟสเดียวกัน ขดลวดที่พันอยู่ที่ฟันของสเตเตอร์ในแต่ละเฟสจะต่ออนุกรมหรือขนานก็ได้ จากรูปที่ 4.4 เป็นการต่อแบบอนุกรม ส่วนโรเตอร์นั้นมีฟัน 4 ซี่ ทั้งโรเตอร์และสเตเตอร์ทำมาจากโลหะซิลิกอน ซึ่งมีสภาพซึมซับทางแม่เหล็กสูงและยอมให้สนามแม่เหล็กจำนวนมากไหลผ่านได้ ฟันของสเตเตอร์ในเฟสเดียวกันจะมีขั้วต่างกันโดยซี่ 1,2,3 เป็นขั้วเหนือและซี่ 1',2',3' เป็นขั้วใต้หลังจากถูกกระตุ้น

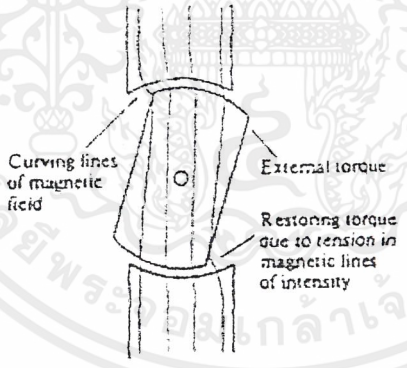
กระแสที่ไหลในแต่ละเฟสถูกควบคุมโดยสวิทช์ ปิด/เปิด ถ้าเฟส 1 ถูกกระตุ้นจะมีกระแสไหลและเกิดฟลักซ์แม่เหล็กดังแสดงในรูปที่ 4.5 แกนโรเตอร์จะอยู่เดียวกับซี่ 1 และ 1' ทำให้ทั้งโรเตอร์และสเตเตอร์อยู่ในแนวเดียวกันกรณีนี้จะทำให้ค่ารีลักแตนซ์มีค่าน้อยที่สุดซึ่งเป็นตำแหน่งที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

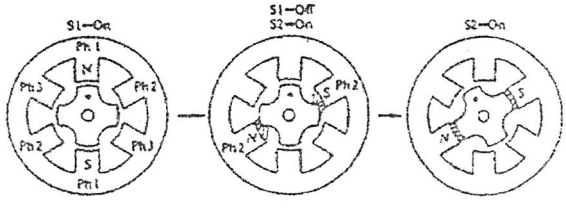


รูปที่ 4.4 แสดงภาพตัดขวางของสเต็ปปิ้งมอเตอร์แบบ 3 เฟส

รูปที่ 4.5 แสดงตำแหน่งสมดุลย์ เมื่อเฟสใดเฟสหนึ่งของสเต็ปปิ้งมอเตอร์ถูกกระตุ้น



รูปที่ 4.6 แสดงแรงภายนอกที่มีผลต่อเส้นแรงแม่เหล็ก



รูปที่ 4.7 แสดงขั้นตอนการเคลื่อนที่ของโรเตอร์เมื่อสเต็ปปิ้งมอเตอร์ถูกกระตุ้น

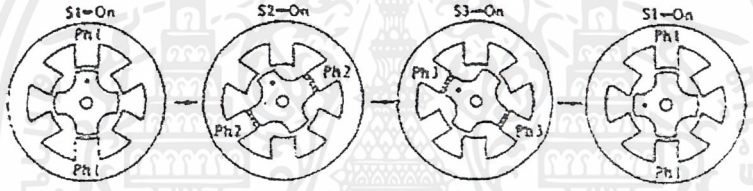
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สมดุลถ้าโรเตอร์ถูกกระทำจากแรงภายนอกจะทำให้เปลี่ยนตำแหน่งดังรูปที่ 4.6 แรงบิดกระทำกับโรเตอร์ในทิศตามเข็มนาฬิกาทำให้ตำแหน่งเปลี่ยนไป มีผลทำให้เส้นแรงแม่เหล็กเคลื่อนที่จากขั้วของโรเตอร์และสเตเตอร์ เมื่อโรเตอร์และสเตเตอร์ไม่ได้อยู่ในแนวเดียวกันแล้วค่ารีลัคแตนซ์จะมีค่ามาก จากนั้นสเต็ปปิ้งมอเตอร์จะทำให้มีค่ารีลัคแตนซ์น้อยที่สุด พอเฟส 2 ถูกกระตุ้น ดังรูปที่ 4.7 โรเตอร์ถูกแรงภายนอกกระทำให้เคลื่อนที่ไป 30 องศาในทิศทวนเข็มนาฬิกา จากนั้นก็จะย้ายจากมุมที่ถูกกระตุ้นกลับไปยังตำแหน่งที่ค่ารีลัคแตนซ์น้อยที่สุด การย้ายจากมุมที่เกิดการกระตุ้นแต่ละครั้งให้กลับไปยังตำแหน่งเดิมเรียกว่าสเต็ป

คุณสมบัติพื้นฐานของสเต็ปปิ้งมอเตอร์แบบวาริเอเบิลรีลัคแตนซ์

1. ช่องว่าง (AIR GAP) ต้องมีขนาดเล็กที่สุด

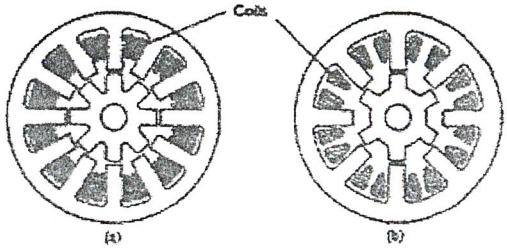
ช่องว่างระหว่างฟันของโรเตอร์และสเตเตอร์ต้องเล็กที่สุดเท่าที่จะเป็นไปได้เพื่อให้ทอร์คที่เกิดขึ้นมีค่ามากและมีความแม่นยำทางตำแหน่งมากขึ้น



รูปที่ 4.8 แสดงขั้นตอนการเคลื่อนที่ของสเต็ปปิ้งมอเตอร์

2. มุมของสเต็ปแคบ

คุณสมบัติอีกประการของสเต็ปคือจะต้องมีมุมของการสเต็ปเล็กที่สุดเท่าที่จะเป็นไปได้ มุมที่แสดงในรูปที่ 4.4 ยังไม่ถือว่าเป็นมุมที่เล็ก แต่รูปที่ 4.9a แสดงมอเตอร์ 3 เฟส ซึ่งมีจำนวนฟันของโรเตอร์และสเตเตอร์เป็น 2 เท่าของรูปที่ 4.4 ส่วนรูปที่ 4.9b แสดงมอเตอร์ 4 เฟส มุมของการสเต็ป 7.5 องศา โดยฟันของโรเตอร์และสเตเตอร์มี 12 และ 16 ซี่ตามลำดับ



รูปที่ 4.9a แสดงมอเตอร์ 3 เฟส รูปที่ 4.9b แสดงมอเตอร์ 4 เฟส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความสัมพันธ์ของมุมของการสเต็ป θ_s, มุมเฟส m, จำนวนซี่ฟันของโรเตอร์ Nr, จำนวนสเต็ป S แสดงดังสมการ

$$S = 360/\theta_s = m.Nr$$

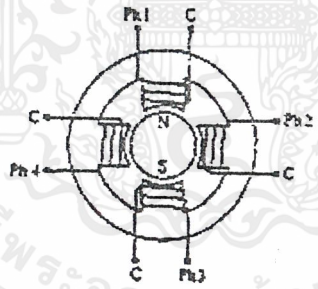
เพื่อที่จะลดขนาดของมุมสเต็ปลงต้องเพิ่งจำนวนซี่ฟันของโรเตอร์โดยที่โครงสร้างของแต่ละขั้วของเฟสใดๆ สเตเตอร์จะมีหลายซี่ฟัน แต่ก็ไม่ใช่องค์ประกอบโดยตรงที่จะกำหนดมุมของสเต็ปปิ้งมอเตอร์

3. การสร้างสเต็ปปิ้งมอเตอร์ให้มีโครงสร้างหลายสเต็ปเพื่อเพิ่มประสิทธิภาพ

โครงสร้างของสเต็ปปิ้งมอเตอร์แบบนี้จะมี 1 เฟสโดยที่โรเตอร์และสเตเตอร์มีซี่ฟันเหมือนกันซึ่งจะเป็นการเพิ่มประสิทธิภาพในด้านทอร์คต่อหน่วยปริมาตรของโรเตอร์

4.5.2 สเต็ปปิ้งมอเตอร์แบบแม่เหล็กถาวร

สเต็ปปิ้งมอเตอร์ชนิดนี้ใช้แม่เหล็กถาวรเป็นโรเตอร์ และมีซี่ฟันของสเตเตอร์ล้อมรอบ ฟันของสเตเตอร์ถูกพันด้วยขดลวดสำหรับสร้างสนามแม่เหล็ก เมื่อต้องการให้สเต็ปปิ้งมอเตอร์แบบแม่เหล็กถาวรมีขนาดของมุมสเต็ปเล็กลงจะต้องเพิ่มขั้วแม่เหล็กของโรเตอร์และจำนวนซี่ฟันของสเตเตอร์ แต่ก็มีขีดจำกัดในการเพิ่มจำนวนขั้วแม่เหล็กของโรเตอร์เนื่องจากการสร้างแม่เหล็กถาวรสร้างโดยมีขั้วแม่เหล็กหลายขั้วทำได้ยาก



รูปที่ 4.10 แสดงโครวสร้างของสเต็ปปิ้งมอเตอร์แบบแม่เหล็กถาวร

ตัวอย่างการทำงานของสเต็ปปิ้งมอเตอร์แบบแม่เหล็กถาวร สมมติว่าสเต็ปปิ้งมอเตอร์แบบแม่เหล็กถาวร ขนาด 4 เฟส มีโรเตอร์เป็นแบบแม่เหล็กถาวรทรงกระบอกและสเตเตอร์มี 4 ซี่ฟันซึ่งรอบๆพันด้วยขดลวด มีรูปแบบพื้นฐานของการทำงานคือ เมื่อสร้างสัญญาณกระตุ้นตามลำดับเฟสโรเตอร์จะหมุนไปตามทิศทางของการกระตุ้น

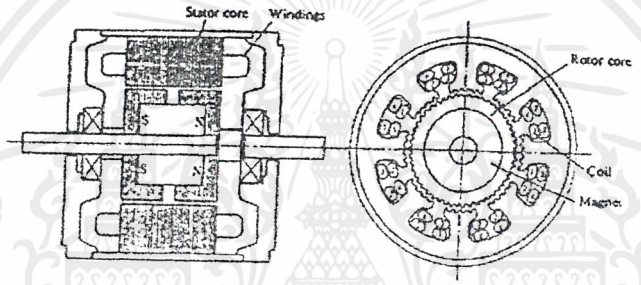
ข้อเสียของสเต็ปปิ้งมอเตอร์แบบแม่เหล็กถาวรคือ มีขนาดมุมสเต็ปใหญ่ทำให้ความละเอียดของสเต็ปต่อรอบน้อยเนื่องจากโครงสร้างของโรเตอร์เป็นแบบแม่เหล็กถาวร การสร้างแม่เหล็กถาวรให้มีขั้วหลายขั้วทำได้ยากทำให้ไม่สามารถสร้างสเต็ปขนาดเล็กได้ สเต็ปปิ้งมอเตอร์แบบแม่เหล็กถาวรส่วนใหญ่จะมีโครงสร้างขนาดเล็ก ทำให้ค่าทอร์คที่ได้ต่อหน่วยปริมาตรต่ำ ถ้าต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปรับปรุงประสิทธิภาพในเรื่องของทอร์ค แม่เหล็กถาวรที่ใช้ต้องทำจากสารแม่เหล็กที่มีสภาพความเป็นแม่เหล็กสูง

4.5.3 สเต็ปปิ้งมอเตอร์แบบไฮบริดจ์

สเต็ปปิ้งมอเตอร์ชนิดนี้มีแกนโรเตอร์เป็นแม่เหล็กถาวร โดยมีการทำงานร่วมกันของมอเตอร์แบบแม่เหล็กถาวรและมอเตอร์แบบวาริเอเบิลรีลัคแตนซ์ได้ ไฮบริดจ์สเต็ปปิ้งมอเตอร์นี้มีโครงสร้างของสเตเตอร์คล้ายโครงสร้างของสเต็ปปิ้งมอเตอร์แบบวาริเอเบิลรีลัคแตนซ์ แต่ต่างกันที่การต่อขดลวดโดยที่แต่ละเฟสของสเต็ปปิ้งมอเตอร์แบบวาริเอเบิลรีลัคแตนซ์จะมีขดลวด 2 ขดแต่ละขดมีขั้วต่างกัน แต่ไฮบริดจ์สเต็ปปิ้งมอเตอร์ขดลวดทั้ง 2 ขดจะพันอยู่ที่ขั้วเดียวกันเรียกว่า ไบโพลาร์(BIPOLAR) ซึ่งในการกระตุ้นแต่ละครั้งจะให้ขั้วที่แตกต่างกัน



รูปที่ 4.11 แสดงโครงสร้างของไฮบริดจ์สเต็ปปิ้งมอเตอร์

คุณสมบัติที่สำคัญของไฮบริดจ์สเต็ปปิ้งมอเตอร์ คือ โครงสร้างของมอเตอร์จะมีแม่เหล็กถาวรอยู่ตรงกลางระหว่างเฟสทั้งสอง การเหนี่ยวนำสนามแม่เหล็กทำได้โดยใช้สนามแม่เหล็กซึ่งสร้างจากสเตเตอร์ซึ่งเป็นสนามแม่เหล็กแบบเฮทเทอโรโพลาร์ (HETEROPOLAR) ดังนั้นทอร์คเกิดจากการทำงานร่วมกันของสนามแม่เหล็ก 2 ชนิดคือ สนามแม่เหล็กจากแม่เหล็กถาวรและสนามแม่เหล็กเหนี่ยวนำที่เกิดจากการกระตุ้นของขดลวดแต่ละขด โครงสร้างของซี่ฟันของสเตเตอร์จะใหญ่กว่าซี่ฟันของโรเตอร์เล็กน้อยเพื่อเพิ่มความถูกต้องแม่นยำทางตำแหน่งของการเคลื่อนที่

หลักการทำงานของไฮบริดจ์สเต็ปปิ้งมอเตอร์ที่แตกต่างจากสเต็ปปิ้งมอเตอร์แบบวารีเอเบิ้ลรีลัคแตนซ์คือ แรงบิดที่เกิดจากสนามแม่เหล็กจะไม่ขึ้นอยู่กับกระแสที่ไหลผ่านขดลวดเพียงอย่างเดียวแต่ขึ้นอยู่กับโครงสร้างของซี่ฟันด้วย ซึ่งซี่ฟันที่ถูกออกแบบเพื่อให้ได้โครงสร้างขนาดเล็ก และใช้แม่เหล็กถาวรเป็นแกนกลางเพื่อลดผลของการเกิดออสซิลเลททางแมคคานิคส์

ข้อดีของไฮบริดจ์สเต็ปปิ้งมอเตอร์คือ มีขนาดสเต็ปขนาดเล็ก มีความละเอียดสเต็ปต่อรอบสูง มีค่าทอร์คสูงกว่าสเต็ปปิ้งมอเตอร์แบบวาริเอเบิ้ลรีลัคแตนซ์ มีแรงเฉื่อยทางแมคคานิคส์น้อยกว่าไฮบริดจ์สเต็ปปิ้งมอเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นอกจากสเต็ปปิ้งมอเตอร์ทั้ง 3 ชนิดที่กล่าวมาแล้วยังสเต็ปปิ้งมอเตอร์ชนิดอื่นๆที่ไม่ได้กล่าวถึงอีกเช่น ลิเนียร์สเต็ปปิ้งมอเตอร์ ซึ่งเป็นสเต็ปปิ้งมอเตอร์ที่ได้รับการออกแบบให้มีการเคลื่อนที่แบบเป็นเชิงเส้น อิเลคโตรดไฮบริดจ์สเต็ปปิ้งมอเตอร์ ซึ่งเป็นสเต็ปปิ้งมอเตอร์กำลังสูงที่ใช้ในงานอุตสาหกรรม เป็นต้น

4.6 วงจรขับ (DRIVE CIRCUITS)

สัญญาณควบคุมที่ใช้สำหรับควบคุมการทำงานของสเต็ปปิ้งมอเตอร์มักจะเป็นสัญญาณที่สร้างจากวงจรดิจิตอล เช่นจากไมโครคอนโทรลเลอร์ ซึ่งเป็นอุปกรณ์จำพวก TTL แรงดันที่ใช้มีค่าเท่ากับ 5 โวลต์ และสามารถจ่ายกระแสได้ไม่มาก แต่เนื่องจากการทำงานของสเต็ปปิ้งมอเตอร์ต้องการแรงดันและกระแสที่สูงกว่านั้น ดังนั้นจึงจำเป็นต้องมีวงจรขับ เพื่อทำหน้าที่จ่ายแรงดันและกระแสที่พอเพียงให้กับตัวสเต็ปปิ้งมอเตอร์ โดยทั่วๆไป วงจรขับมักจะสร้างจากไบโพลาร์ทรานซิสเตอร์ที่นำมาต่อใช้งานเป็นสวิตซ์ปิด/เปิด ให้กระแสไหลผ่านไปยังขดลวดในทิศทางเดียว เราเรียกวงจรขับแบบนี้ว่า ยูนิโพลาร์ การจ่ายกระแสเพียงทิศทางเดียว แต่ถ้าใช้สเต็ปปิ้งมอเตอร์แบบไฮบริดจ์หรือแบบแม่เหล็กถาวร ซึ่งมักจะมีสองเฟส จะต้องใช้วงจรขับที่สามารถจ่ายกระแสได้สองทิศทาง เรียกวงจรขับแบบนี้ว่า ไบโพลาร์ ซึ่งประกอบด้วย ทรานซิสเตอร์หลายตัวต่อเป็นวงจรแบบบริดจ์

4.6.1 วงจรขับแบบยูนิโพลาร์ (UNIPOLAR DRIVE CIRCUIT)

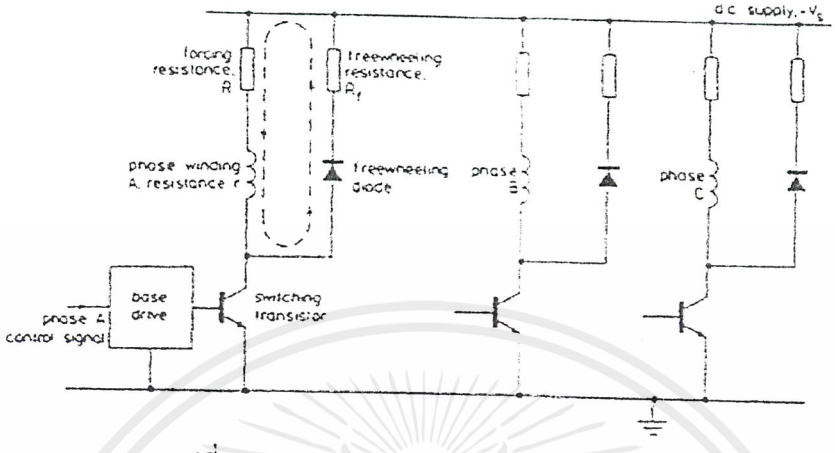
วงจรขับพื้นฐานที่เหมาะสมสำหรับสเต็ปปิ้งมอเตอร์แบบวาริเอเบิ้ลรีลัคแตนซ์ดังที่แสดงไว้ในรูปที่ 4.12 ขดลวดแต่ละเฟสจะถูกกระตุ้นโดยวงจรขับแต่ละชุด ซึ่งวงจรขับแต่ละชุดก็จะได้รับสัญญาณควบคุมจากไมโครคอนโทรลเลอร์

กระแสจะไหลผ่านขดลวดแต่ละเฟสเมื่อสวิตช์ซึ่งทรานซิสเตอร์อยู่ในสภาวะอิ่มตัวเนื่องจากกระแสที่ไบอัสทางด้านเบส ในสภาวะเช่นนี้แรงดันดีซีจากแหล่งจ่ายไฟจะไหลผ่านตัวต้านทานผ่านขดลวดของสเต็ปปิ้งมอเตอร์ และไหลผ่านทรานซิสเตอร์ เนื่องจากแรงดันตกคร่อมทรานซิสเตอร์ในสภาวะอิ่มตัวมีค่าน้อย (ประมาณ 0.1 โวลต์) แรงดันจากแหล่งจ่ายไฟทั้งหมดจะทำให้เกิดกระแสไหลผ่านขดลวดโดยมีความสัมพันธ์ กับผลรวมของความต้านของขดลวด (r) และความต้านทานตัวต้านทาน (R) ดังนี้

$$V_s = i.(r+R)$$

ตามปกติขดลวดของสเต็ปปิ้งมอเตอร์จะแสดงคุณสมบัติของตัวเหนี่ยวนำ (L) ซึ่งทำให้การตอบสนองต่อกระแสที่ไหลผ่านตัวมันช้า ส่งผลต่อการทำงานของมอเตอร์ที่ความเร็วสูง การใส่ตัวต้านทานอนุกรมเข้าไว้กับขดลวดจะช่วยแก้ปัญหานี้ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

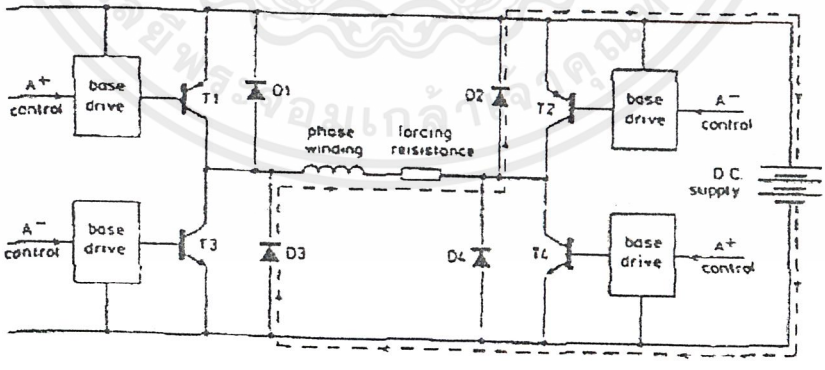


รูปที่ 4.12 วงจรขับแบบยูนิโพลาร์

ความเหนี่ยวนำของขดลวดยังทำให้กระแสไม่หยุดไหลทันทีทันใดที่ทรานซิสเตอร์อยู่ในสถานะปิด ทำให้เกิดแรงดันเหนี่ยวตกคร่อมคอลเล็คเตอร์และอิมิตเตอร์ ซึ่งเป็นสาเหตุสำคัญที่ทำให้วงจรขับเสียหาย ปัญหานี้แก้ไขได้โดยการต่อ ฟรีวิลลิ่งไดโอด (FREEWHEELING DIODE) และฟรีวิลลิ่งรีซิสเตอร์ (FREEWHEELING RESISTANCE) เพื่อเป็นทางผ่านของกระแสแทน

4.6.2 วงจรขับแบบไบโพลาร์ (BIPOLAR DRIVE CIRCUIT)

วงจรขับชนิดนี้จะต่อทรานซิสเตอร์เป็นแบบบริดจ์ วงจรบริดจ์หนึ่งชุดจะขับมอเตอร์ได้หนึ่งเฟสเหมาะที่ใช้กับสเต็ปปิ้งมอเตอร์แบบแม่เหล็กถาวร หรือแบบไฮบริดจ์ ดังแสดงในรูปที่ 4.13



รูปที่ 4.13 วงจรขับแบบไบโพลาร์

ทรานซิสเตอร์จะพลัดกันทำงานทีละคู่ตามทิศทางของกระแสที่ต้องการ สำหรับการกระตุ้นขดลวดในทิศบวก ทรานซิสเตอร์ T1 และ T4 จะทำงาน ทำให้เกิดทางเดินของกระแสจากแหล่งจ่ายไฟ ไหลผ่านทรานซิสเตอร์ T1 จากนั้นไหลผ่านขดลวดของมอเตอร์ผ่านตัวต้านทานแล้วจึงไหลเข้าทรานซิสเตอร์ T4 กลับเข้าสู่แหล่งจ่ายไฟ ในทางกลับกันในกรณีการจ่ายกระแสในทิศลบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(ทิศตรงกันข้าม) ทรานซิสเตอร์ T2 และ T3 จะทำงาน เพื่อให้กระแสไหลผ่านขดลวดในทิศตรงกันข้าม

จากรูปจะเห็นว่ามีไดโอดสี่ตัว ต่อขนานกับทรานซิสเตอร์แต่ละตัว จุดประสงค์เพื่อให้เกิดเส้นทางไหลของกระแสฟรีวิลลิ่ง (FREEWHEELING CURRENT) บริเวณไดโอด D2 และ D3 จะเป็นทางผ่านของกระแสหลังจากที่ทรานซิสเตอร์ T1 และ T4 หยุดทำงาน ส่วน D1 และ D4 จะมีกระแสไหลผ่านขณะที่ทรานซิสเตอร์ T2 และ T3 หยุดทำงาน กระแสฟรีวิลลิ่งในวงจรขับแบบไบโพลาร์จะสิ้นสุดเร็วกว่าแบบยูนิโพลาร์ เพราะฉะนั้นจึงไม่จำเป็นต้องต่อฟรีวิลลิ่งรีซิสแตนซ์เหมือนยูนิโพลาร์

4.7 การกระตุ้นและควบคุมการหมุนของสเต็ปปิ้งมอเตอร์

การกระตุ้นและควบคุมการหมุนของมอเตอร์ให้เคลื่อนที่ไปแต่ละสเต็ป ทำได้โดยจ่ายกำลังไฟฟ้าไปยังขดลวดแต่ละขดบนสเตเตอร์ ซึ่งต้องป้อนเป็นแบบซีเควนเชียลในรูปแบบที่ถูกต้อง แบ่งได้เป็น 3 รูปแบบ คือแบบเวฟ (wave) , แบบ 2 เฟส และแบบครึ่งสเต็ป(half step) ทั้ง 3 แบบต่างก็มีข้อดีและข้อเสียแตกต่างกันออกไป

แบบเวฟเป็นการกระตุ้นที่ง่ายที่สุด โดยทำการกระตุ้นขดลวดทีละขดในเวลาหนึ่งและเรียงถัดกันไป ดังเช่นขดที่ 1,2,3,4,1 หรือ 1,4,3,2,1 ขึ้นอยู่กับทิศทางที่ต้องการให้หมุน ดังนั้นจึงมีขดลวดเพียงขดเดียวในเวลาที่ถูกกระตุ้นเท่านั้น วงจรกระตุ้นแบบเวฟจึงมีราคาถูกและง่าย ขั้นตอนการทำงานต่างๆ แสดงดังในตารางที่ 4.2

สเต็ปที่	เฟสที่ 1	เฟสที่ 2	เฟสที่ 3	เฟสที่ 4
1	ทำงาน	-	-	-
2	-	ทำงาน	-	-
3	-	-	ทำงาน	-
4	-	-	-	ทำงาน

ตารางที่ 4.2 แสดงขั้นตอนการกระตุ้นขดลวดแต่ละเฟสแบบเวฟ

แบบ 2 เฟสเป็นการกระตุ้นอีกรูปแบบหนึ่งซึ่งคล้ายกับแบบเวฟแต่การกระตุ้นแบบนี้จะทำการโดยจ่ายกำลังไฟฟ้าไปที่ขดลวด 2 ขด ที่อยู่ใกล้กันในเวลาเดียวกันและเรียงถัดกันไปเช่นเดียวกับการเวฟคือ ขดลวดที่ถูกกระตุ้น 12,23,34,41,12 หรือ 14,43,32,21,14 ขึ้นอยู่กับทิศทางการหมุน การเพิ่มจำนวนของขดลวดที่ถูกกระตุ้นนี้ทำให้เพิ่มแรงบิดได้มากกว่าแบบเวฟ โรเตอร์จะเคลื่อนที่ได้ด้วยแรงดึงแบบเต็มแรง จาก 2 ขดลวดที่ถูกกระตุ้นพร้อมกันและต่อไปด้วยแรงดึงจากอีก 2 ขด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลวดถัดไป สำหรับข้อเสียคือ การกระตุ้นแบบนี้ต้องใช้แหล่งจ่ายกำลังไฟฟ้ามากขึ้น ขั้นตอนการทำงานต่างๆแสดงดังในตารางที่ 4.3

สเต็ปที่	เฟสที่ 1	เฟสที่ 2	เฟสที่ 3	เฟสที่ 4
1	ทำงาน	ทำงาน	-	-
2	-	ทำงาน	ทำงาน	-
3	-	-	ทำงาน	ทำงาน
4	ทำงาน	-	-	ทำงาน

ตารางที่ 4.3 แสดงขั้นตอนการกระตุ้นขดลวดแต่ละเฟสแบบ 2 เฟส

แบบครึ่งสเต็ปเป็นรูปแบบที่เกิดจากการผสมผสานระหว่างการกระตุ้นแบบเวฟและแบบ 2 เฟส เพื่อเพิ่มจำนวนสเต็ปต่อรอบอีกเท่าตัวหนึ่ง ในระบบนี้จะทำการกระตุ้นขดลวดเรียงกันไปเป็นลำดับนี้ ขดลวดที่ถูกกระตุ้น 1,12,2,23,3,34,4,41,1 หรือในการหมุนอีกทิศทางหนึ่งจะได้เป็น 1,14,4,43,3,32,2,21,1 แรงบิดที่ได้จากการกระตุ้นแบบนี้จะเพิ่มมากขึ้นอีก เพราะช่วงสเต็ปมีระยะสั้นลงและแต่ละสเต็ปเกิดแรงดึงจากขดลวด 2 ขดที่ถูกกระตุ้นพร้อมกัน ความถูกต้องของตำแหน่งมีเพิ่มมากขึ้นแต่ต้องพึงระวังไว้อีกประการหนึ่งว่าเมื่อถูกกระตุ้นให้ทำงานในรูปแบบนี้จะต้องทำการหมุนถึง 2 สเต็ปจึงจะได้เท่ากับ 2 แบบแรก สำหรับแหล่งจ่ายกำลังไฟฟ้าต้องใช้เทียบเท่ากับแบบ 2 เฟสจึงจะเพียงพอขั้นตอนการทำงานต่างๆดังแสดงในตารางที่ 4.4

สเต็ปที่	เฟสที่ 1	เฟสที่ 2	เฟสที่ 3	เฟสที่ 4
1	ทำงาน	-	-	-
2	ทำงาน	ทำงาน	-	-
3	-	ทำงาน	-	-
4	-	ทำงาน	ทำงาน	-
5	-	-	ทำงาน	-
6	-	-	ทำงาน	ทำงาน
7	-	-	-	ทำงาน
8	ทำงาน	-	-	ทำงาน

ตารางที่ 4.4 แสดงขั้นตอนการกระตุ้นขดลวดแต่ละเฟสแบบครึ่งสเต็ป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.8 การทำงานของวงจรควบคุมสเต็ปปิ้งมอเตอร์

การทำงานของวงจร Drive เริ่มจากพอร์ต printer ของคอมพิวเตอร์ส่งผ่านตามสายสัญญาณมาที่วงจร Drive เข้ามาทางขาเบสของทรานซิสเตอร์ TIP 41 ซึ่งทรานซิสเตอร์ตัวนี้ทำหน้าที่เสมือนเป็นสวิตช์ตัดต่อ เมื่อมีสัญญาณมาเข้าที่ขาเบสแล้ว จะทำให้ทรานซิสเตอร์อิ่มตัว กระแสไฟฟ้าที่มาคอยอยู่ที่ขาคอลเลคเตอร์จึงสามารถไหลไปขาอีมิตเตอร์โดยผ่านขดลวด 1 ขดภายในสเต็ปปิ้งมอเตอร์ แล้วลงกราวน์ได้

การกระตุ้นการทำงานของสเต็ปปิ้งมอเตอร์จะต้องเป็นไปตามลำดับ (sequence) ทั้งในทิศทางตามเข็มนาฬิกาและทวนเข็มนาฬิกา คือเราจะใช้โปรแกรม Delphi เขียนโปรแกรมติดต่อกับ port printer โดยใช้ address port 378H เป็น port เอาท์พุท ซึ่งมีบิตทั้งหมด 8 บิต เราจึงนำสัญญาณจากแต่ละบิตมาเป็นสัญญาณป้อนเข้าขาเบสของทรานซิสเตอร์ TIP 41 ที่ต่อกับขดลวดแต่ละขดของสเต็ปปิ้งมอเตอร์ทั้ง 2 ตัว (ทั้งแกน X และ Y) และการต่อจะต้องต่อตามลำดับการทำงานของขดลวดที่ถูกต้อง มิเช่นนั้นมอเตอร์จะหมุนไม่เป็นลำดับ ถ้าเป็นสเต็ปปิ้งมอเตอร์แบบ 6 สาย จะมีสายไฟขดลวด 4 เส้น และ Centertap 2 เส้นส่วนในวงจรนี้จะใช้ Centertap ต่อลงกราวน์ ส่วนอีก 4 เส้นจะต่อเข้ากับขาอีมิตเตอร์ของทรานซิสเตอร์ TIP 41 ของแต่ละชุดโดยสเต็ปปิ้งมอเตอร์ 1 ตัวจะต้องใช้ชุดของทรานซิสเตอร์ไดรฟ์ 4 ชุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
ทฤษฎีเกี่ยงกับ PORT PRINTER

5.1 รายระเอียดเกี่ยวกับ SLOT มาตรฐานของ IBM PC

SA0-SA9 (อินพุต/เอาท์พุต) เป็นแอดเดรสของระบบที่ใช้ติดต่อกับหน่วยความจำและอุปกรณ์อินพุต/เอาท์พุต สายสัญญาณนี้จะต่อกับหน่วยความจำได้ 1 MB แต่ถ้าต้องการเชื่อมขยายแอดเดรสจะต้องใช้สายแอดเดรส LA17-LA23 การใช้สัญญาณ SA0-SA19 จะต้องแอคทีฟขณะที่สัญญาณ BALE เป็น 1 และจะแลตช์ไปใช้ในขณะเปลี่ยนจาก 1 ไป 0 สัญญาณ BALE เป็นสัญญาณที่มาจากไมโครโปรเซสเซอร์ หรือ ดีเอ็มเอคอนโทรเลอร์

CLK (เอาต์พุต) เป็นสัญญาณนาฬิกาของระบบ ในกรณีที่ไอบีเอ็มที จะส่งสัญญาณนี้เป็นสัญญาณขนาด 6 MHZ โดยมีช่วงเวลาประมาณ 167 นาโนวินาที สัญญาณเป็นรูปสี่เหลี่ยมมี DUTY CYCLE 50 เปอร์เซ็นต์ สัญญาณนี้มีจุดมุ่งหมายเพื่อใช้ในการซิงโครนัสระบบ มิได้มีจุดหมายสำหรับใช้เป็นฐานเวลา

RESET DRV (เอาต์พุต) สัญญาณนี้ใช้สำหรับรีเซตระบบในขณะเปิดเครื่อง หรือขณะที่แหล่งจ่ายไฟที่ขาด หรือไฟตก สัญญาณนี้จะแอคทีฟเมื่อลอจิกเป็น 1

SD0-SD7 (อินพุต/เอาต์พุต) เป็นสัญญาณข้อมูลขนาด 8 บิท ที่ใช้ติดต่อกับหน่วยความจำไมโครโปรเซสเซอร์ และอุปกรณ์อินพุต/เอาต์พุต บิต D0 เป็นบิตที่มีนัยสำคัญน้อยที่สุด

BALE (เอาต์พุต) เป็นสัญญาณที่ใช้สำหรับการแลตช์แอดเดรสของระบบ สัญญาณนี้จะมาจาก 82288 ตัวควบคุมบัสสัญญาณที่จะใช้แลตช์แอดเดรสเมื่อเปลี่ยนจาก 1 กับ 0 และสัญญาณนี้จะได้รับการทำให้เป็น 1 ขณะที่กำลังทำ ดีเอ็มเอ

I/O CHK (อินพุต) สัญญาณตรวจสอบของอินพุต/เอาต์พุต เพื่อบอกข้อมูลกับระบบเช่นเดียวกับการตรวจสอบพาริตี้ ดังนั้นถ้าบนอินพุต/เอาต์พุตมีข้อผิดพลาด สัญญาณนี้จะแอคทีฟเพื่อให้ส่งสัญญาณเตือนในลักษณะ PARITY ERROR

I/O CHRDY (อินพุต) สัญญาณนี้จะได้รับการทำให้เป็น 0 ด้วยหน่วยความจำหรืออุปกรณ์อินพุตเอาต์พุต การใช้สัญญาณนี้ก็เพื่อใช้อุปกรณ์อินพุต/เอาต์พุตที่ช้าจะได้ติดต่อกับระบบด้วยการส่งสัญญาณมายัง ซีพียู เพื่อซิงโครนัสระบบได้

IRQ3-IRQ7 (อินพุต) เป็นสัญญาณอินเตอร์รัพต์

IOR (อินพุต/เอาต์พุต) สัญญาณอินพุต เอาต์พุตเป็นสัญญาณที่ส่งมาจากซีพียู การควบคุมสัญญาณนี้มาจาก 80286 และดีเอ็มเอ คอนโทรลเลอร์ สัญญาณนี้ แอคทีฟ 0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SMEMR (เอาต์พุต) MEMR (อินพุต/เอาต์พุต) สัญญาณนี้เป็นสัญญาณควบคุมการอ่านข้อมูลจากหน่วยความจำ SMEMR ใช้สำหรับติดต่อกับหน่วยความจำในส่วน 1 MB แรก หรือถอดรหัสมาจากแอดเดรสส่วนล่าง ส่วน MEMR นี้แอคทีฟกับหน่วยความจำได้หมด 16 MB

DRQ0-DRQ3 (อินพุต) สัญญาณการขอดีเอ็มเอแชนแนล 0-3 โดยสัญญาณนี้จะมาจากอุปกรณ์อินพุต/เอาต์พุต

ALEN (เอาต์พุต) อีนาเบิลแอดเดรส เป็นสัญญาณที่ใช้สำหรับการแยกบัสแอดเดรสในการทำดีเอ็มเอ เมื่อสัญญาณนี้แอคทีฟ แสดงว่าดีเอ็มเอ คอนโทรลเลอร์ควบคุมการทำงานทางแอดเดรสแทนซีพียู

REFRESH (อินพุต/เอาต์พุต) เป็นสัญญาณที่ใช้ในการแสดงสัญญาณรีเฟรชไซเคิล สัญญาณนี้ส่งมาจากไมโครโปรเซสเซอร์ผ่านทางช่องอินพุต/เอาต์พุต

T/C (เอาต์พุต) สัญญาณ TERMINAL COUNT เป็นสัญญาณพัลซ์เมื่อดีเอ็มเอนับจำนวนมาครบตามที่กำหนด

MEM CS16 (อินพุต) สัญญาณนี้ส่งมาจากเมนบอร์ด ถ้าหากการถ่ายเทข้อมูลต้องการสถานะรอ

IO CS16 (อินพุต) สัญญาณนี้เป็นตัวส่งมาจากเมนบอร์ดว่าอินพุต/เอาต์พุต ต้องการสถานะรอ

OSC (เอาต์พุต) สัญญาณนาฬิกา 70 นาโนวินาที หรือ ประมาณ 1431818 MHz

OWS (อินพุต) เป็นสัญญาณที่บอกซีพียูว่าการทำงานหนึ่งรอบของบัสไม่จำเป็นต้องการแทรกสถานะรอ

Number Port(Hex)	Type of Equipment
001-01F	DMA COBTROLLER 1,8237A-5
020-03F	INTERRUPT CONTROLLER หมายเลข 1,8259A ตัวหลัก
040-05F	TIMER 8254-2
060-06F	8042 KEYBOARD
070-07F	CLOCKNMC-MOSRAM
080-09F	DMA PAGEREGKHER
0A0-0BF	INTERRUPT CONTROLLER 28237A-5
0C0-0DF	DMA COBTROLLER 28237A-5
0	CLEAR MATH PROCESSOR

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

0F	RESET MATH PROCESSOR
0F8-0FF	MATH PROCESSOR
110-118	HARD DISK
200-207	IO GAME
208-277	NOT USE
278-27F	INTERIOR PORT NO 2
280-2F7	NOT USE
2F8-2FF	SERAL PORT NO 2
300-31F	PROTOTYPE CARD
320-35F	NOT USE
360-36F	SPARE
370-377	NOT USE
378-37F	PRINTER PORT NO 1
380-38F	SDLC,BKYNC 2
390-3AF	NOT USE
3B0-3BF	MONOCHROM AND PRINTER
3C0-3CF	SPARE
3D0-3DF	COLOUR DISPLAY MONITOR
3F0-3F7	CONTROL DISKKET
3F8-3FF	SERAL PORT NO 2

ตารางที่ 5.1 ตารางการแสดงแอดเดรสพอร์ทของ IBM PC

5.2 การจัดแอดเดรสสำหรับหน่วยความจำและ I/O

5.2.1 การจัดแอดเดรสของพอร์ต I/O ใน IBM/PC

ในการควบคุมการตรวจสอบสภาวะในการทำงาน รวมทั้งการอ่านข้อมูลจากอุปกรณ์ที่เป็นชิพซัพพอร์ทหรือการ์ดในการควบคุมต่างๆที่ใช้ในระบบของ IBM/PC จะกะทำโดยผ่านทางพอร์ต I/O ของระบบด้วย และเนื่องจากการควบคุมหรือติดต่อกับพอร์ทเหล่านี้ต้องการกระทำโดยการอ้างถึงแอดเดรสของพอร์ต I/O เหล่านั้นโดยตรง เราจึงจำเป็นต้องศึกษาถึงหลักการอ้างแอดเดรสของ 8088 ในเครื่อง PC ด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับแอดเดรสที่จัดไว้สำหรับ I/O นั้น จะเป็นแอดเดรสที่ถูกสร้างขึ้นโดย 8088 ซึ่งแอดเดรสเหล่านี้เป็นแอดเดรสที่จัดไว้สำหรับ I/O พอร์ต โดยเฉพาะ คือแยกจากแอดเดรสของหน่วยความจำโดยเด็ดขาดส่วนการส่งข้อมูลให้กับพอร์ตเหล่านี้ต้องกระทำโดยการใช้คำสั่ง OUT ของ 8088 ส่งข้อมูลนั้นไปยังแอดเดรสของพอร์ตที่ต้องการ และสำหรับการตรวจสอบหรือการอ่านข้อมูลจากพอร์ต ก็จะทำได้โดยการใช้คำสั่ง IN ของ 8088 อ่านข้อมูลจากแอดเดรสของพอร์ตที่ต้องการเช่นกัน

ภายในไมโครโปรเซสเซอร์ 8088 จะมีแอดเดรสสำหรับการใช้พอร์ต I/O อยู่ทั้งสิ้น 65,536 แอดเดรส ซึ่งทำให้การอ้างแอดเดรสของพอร์ต I/O ที่ทำงานร่วมกับ 80288 นั้น ต้องใช้จำนวนเส้นแอดเดรสในบัสแอดเดรสทั้งสิ้น 16 เส้น แต่สำหรับใน IBM/PC นี้ถูกออกแบบมาให้ใช้แอดเดรสเฉพาะ 10 เส้นล่าง คือ A0-A9 เท่านั้น ดังนั้นในการอ้างอิงถึงแอดเดรสของพอร์ตของอุปกรณ์หรือชิพพอร์ตใดๆที่ใช้ร่วมกับ IBM/PC จึงใช้สันจำนวนแอดเดรสเพียง 10 เส้น ด้วย โดยเส้นแอดเดรสที่เหลือคือ A0-A15 นั้น จะไม่ถูกนำไปใช้งานแต่แอดเดรสบนเส้นแอดเดรสเหล่านี้ยังคงเปลี่ยนแปลงตามค่าแอดเดรสพอร์ตที่กำหนดไว้ในคำสั่ง OUT หรือ IN อยู่ด้วย เพียงแต่ไม่ได้ถูกนำมาดีโคดร่วมกับแอดเดรส 0010H นั้น จะให้ผลเหมือนกับการส่งข้อมูลไปยังพอร์ตที่ตรงกับแอดเดรส 04100H,0810H,0C10H ทั้งนี้เนื่องจากแอดเดรส 6 บิทบนไม่ถูกนำมาใช้งาน จึงทำให้การเปลี่ยนแปลงค่าแอดเดรสบนเส้นแอดเดรส A0-A15 นั้นไม่ทำให้เกิดความแตกต่างใดๆขึ้น

เนื่องจากใน IBM/PC ได้ใช้งานเส้นแอดเดรสเพียง 10 เส้น ดังนั้นจึงสามารถที่จะอ้างแอดเดรสของพอร์ตได้สูงสุดเพียง 1024 เท่านั้น นอกจากนี้ในกรณีที่เป็นการอ่านข้อมูลจากพอร์ตของ IBM/PC ข้อมูลที่บิท A9 เป็น 0 แล้ว เราจะทำการอ่านข้อมูลได้เฉพาะจากพอร์ตของอุปกรณ์หรือชิพพอร์ตต่างๆที่อยู่บนเมนพอร์ด ของ IBM/PC เช่น 8253-5,8237-5,8259A เท่านั้น แต่ถ้าข้อมูลในบิท A9 นี้เป็น 1 ก็จะทำการอ่านข้อมูลได้เฉพาะจากพอร์ตที่อยู่บนการ์ดต่างๆเท่านั้น

พอร์ตบน IBM/PC ทั้ง 1024 พอร์ต ถูกแบ่งออกเป็น 2 กลุ่ม โดยที่กลุ่มแรกเป็นกลุ่มของพอร์ตที่อยู่บนเมนบอร์ด และกลุ่มที่สองปีนกลุ่มที่จัดเตรียมไว้สำหรับพอร์ตที่อยู่บนการ์ดต่างๆ

สำหรับในกรณีของการส่งข้อมูลให้กับพอร์ตทั้ง 1024 พอร์ต เราสามารถที่จะเลือกส่งไปยังพอร์ตใดๆใน IBM/PC ได้ ดังนั้นการเลือกแอดเดรสสำหรับพอร์ตที่อยู่บนการ์ดจึงสามารถทำได้โดยสะดวก แต่อย่างไรก็ตามสิ่งหนึ่งที่จะต้องคำนึงถึงก็คือ ถ้าแอดเดรสที่เราเลือกให้กับพอร์ตนี้ตรงกับค่าแอดเดรสเดิมที่มีอยู่บนบอร์ด เมื่อเราทำการส่งข้อมูลให้กับพอร์ตที่อยู่ในตำแหน่งแอดเดรสนี้ก็จะเท่ากับเป็นการส่งข้อมูลให้กับพอร์ตที่อยู่บนการ์ดด้วย ซึ่งในกรณีเช่นนี้อาจก่อให้เกิดความผิดพลาดขึ้นเช่นกัน ดังนั้นในการกำหนดค่าแอดเดรสที่ถูกสร้างขึ้นบนการ์ดน่างๆจึงควรใช้ค่าแอดเดรสที่แอดเดรสบิท A9 มีค่าเป็น 1 คือ แอดเดรส 0FE00H คือ 0FFFFH เท่านั้น

5.2.2 การใช้งานแอดเดรสในพอร์ต I/O ของ IBM/PC

จากที่กล่าวมาแล้ว พอร์ต I/O ทั้ง 1024 พอร์ต ใน IBM/PC ซึ่งถูกแบ่งออกเป็น 2 กลุ่มๆละ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

512 พอร์ต สำหรับในหัวข้อ นี้จะกล่าวถึงการใช้งานพอร์ตต่างๆ โดยจะแบ่งออกเป็น 2 กลุ่มตามที่ได้อธิบายไว้ตามที่ผ่านมาดังนี้

1.ในกลุ่มแรกนี้เป็นของพอร์ต I/O ที่อยู่บนเมนบอร์ดของ IBM/PC ซึ่งจะมีแอดเดรสอยู่ที่ตำแหน่ง 000H จนถึง 01FF หรือแอดเดรสที่มีบิท A9 เป็น 0 นั่นเอง

สำหรับแอดเดรสของพอร์ต I/O ในกลุ่มนี้จะถูกใช้ในการอ้างแอดเดรสของชิพซัพพอร์ทและอุปกรณ์ I/O ต่างๆบนเมนของ IBM/PC จะเห็นว่าแอดเดรส 00C0H จนถึงแอดเดรส 01FFH นั้นไม่ได้ถูกใช้งานบนเมนบอร์ด ดังนั้นในกรณีดังกล่าวเราก็สามารถที่จะใช้งานแอดเดรสต่างๆเหล่านี้ได้ อย่างไรก็ตามแอดเดรสเหล่านี้ยังคงถูกดีโคดให้เป็นแอดเดรสที่ใช้ในการอ่านข้อมูลจากพอร์ต I/O บนเมนบอร์ดเท่านั้น ดังนั้นการใช้ค่าแอดเดรส 00C0H-01FFH กับพอร์ต I/O บนการ์ดหรือบนพอร์ตที่เราสร้างขึ้นนั้น ต้องเป็นพอร์ตเอาต์พุตเพียงชนิดเดียวเท่านั้น กล่าวคือ จะทำการอ่านข้อมูลที่มีค่าแอดเดรสอยู่ในช่วง 00C0H-01FFH ไม่ได้

2. ในกลุ่มที่ 2 นี้ จะเป็นกลุ่มของพอร์ต I/O ที่ถูกใช้งานบนการ์ดที่เสียบใช้งานบน slot ต่างๆของ IBM/PC สำหรับแอดเดรสของพอร์ตเหล่านั้นจะเริ่มต้นจากแอดเดรส 0200H-03FFH ซึ่งก็คือแอดเดรสที่มีบิท A9 เป็น 1 นั่นเอง สำหรับการใช้งานแอดเดรสของพอร์ต I/O ในกลุ่มนี้

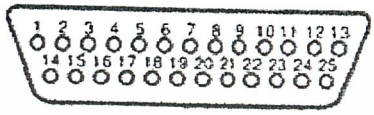
อย่างไรก็ตามการใช้งานแอดเดรสในกลุ่มนี้อาจจะเปลี่ยนแปลงได้ ทั้งนี้ขึ้นอยู่กับการใช้งานบนการ์ดต่างๆร่วมกับ IBM/PC โดยการ์ดที่ถูกออกแบบผลิตขึ้นใหม่นั้นอาจจะใช้ค่าแอดเดรสต่างๆที่เหลืออยู่นี้ได้ ดังนั้นก่อนที่จะทำการออกแบบวงจรอินเทอร์เฟสที่จำเป็นต้องใช้แอดเดรสสำหรับพอร์ต I/O จึงควรจะตรวจสอบดูก่อนว่าการ์ดต่างๆที่ใช้อยู่ในระบบของ IBM/PC ที่เราใช้อยู่นั้นมีการ์ดใดบ้างและการ์ดเหล่านั้นใช้งานแอดเดรสใดบ้าง จากนั้นจึงทำการออกแบบวงจรอินเทอร์เฟสโดยเลือกใช้เฉพาะแอดเดรสที่ยังไม่ถูกใช้งาน

5.3 พอร์ตขนาน

พอร์ตขนานหรือพอร์ตเครื่องพิมพ์ของคอมพิวเตอร์มีขั้วต่อเป็นคอนเนกเตอร์แบบ D ขนาด 25 ขา (DB-25)มีการจัดขาสัญญาณดังรูปที่ 5.1 และมีหน้าที่การทำงานแสดงในตารางที่ 2 ภายในพอร์ตขนานประกอบด้วยรีจิสเตอร์พื้นฐานที่ใช้ในการรับและส่งข้อมูล 3 ตัว คือ รีจิสเตอร์ DATA,STATUS และ CONTROL โดยแอดเดรสของรีจิสเตอร์ทั้งสามนี้จะมีตำแหน่งไล่เรียงกันไปตามลำดับขึ้นอยู่กับแอดเดรสของพอร์ตขนานเช่น ที่พอร์ตขนาน LPT1 แอดเดรสของรีจิสเตอร์ DATA อยู่ที่ &H378 (&H เป็นตัวอักษรที่แสดงว่าข้อมูลนี้เป็นข้อมูลเลขฐานสิบหก) ในขณะที่แอดเดรสของรีจิสเตอร์ STATUS จะอยู่ที่ &H379 และแอดเดรสของรีจิสเตอร์ CONTROL จะอยู่ที่ &H37A โดยความสัมพันธ์ระหว่างแอดเดรสพอร์ตขนานกับแอดเดรสของรีจิสเตอร์พื้นฐานทั้งสามตัวแสดงในตารางที่ 5.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตำแหน่งของพอร์ตขนานส่วนใหญ่จะมีตำแหน่งพอร์ตเริ่มต้นอยู่ที่ &H378 แต่ก็ไม่ใช่ตำแหน่งสำหรับคอมพิวเตอร์ทุกเครื่อง ดังนั้นวิธีการง่ายๆเพื่อตรวจสอบตำแหน่งของพอร์ตขนานที่ใช้งานอยู่ สามารถทำได้โดยการใช้โปรแกรม DEBUG แสดงค่าของหน่วยความจำตำแหน่ง 0040:08 ซึ่งใช้สำหรับเก็บค่าแอดเดรสของพอร์ตขนาน LPT1,LPT2 และ LPT3 เอาไว้ดังตัวอย่างกรอบแยกที่ 1



รูปที่ 5.1 รูปร่างและตำแหน่งขาของคอนเนกเตอร์พอร์ตขนาน DB-25 ตัวเมีย(มองจากด้านหน้า)

กรอบแยกที่ 1

```
C:\dos\debug
-d 0040 : 08
0040 : 0008  78 03 78 02 00 00 00 00
```

จากตัวอย่างข้างต้นพอร์ต LPT1 มีแอดเดรสอยู่ที่ &H378 ส่วนพอร์ต LPT2 มีแอดเดรสอยู่ที่ &H278 สำหรับ LPT3 และ LPT4 นั้นยังไม่ได้ถูกกำหนดเอาไว้ นอกจากการใช้โปรแกรม DEBUG แล้วยังสามารถใช้โปรแกรม MSD.EXE ของ DOS เพื่อตรวจสอบได้อีกด้วย

จากตารางที่ 5.3 สามารถแบ่งคุณสมบัติของพอร์ตขนานได้ 2 รูปแบบคือ พอร์ตที่ทำหน้าที่เป็นเอาต์พุต และพอร์ตที่ทำหน้าที่เป็นอินพุต ดังมีรายละเอียดดังต่อไปนี้

ขา	หน้าที่	พอร์ต		ทิศทาง
		ชื่อ	บิต	
1	$\overline{\text{Strobe}}$	Control	C0	เอาต์พุท
2	Data บิต 0	Data	D0	เอาต์พุท
3	Data บิต 1	Data	D1	เอาต์พุท
4	Data บิต 2	Data	D2	เอาต์พุท
5	Data บิต 3	Data	D3	เอาต์พุท
6	Data บิต 4	Data	D4	เอาต์พุท
7	Data บิต 5	Data	D5	เอาต์พุท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8	Data บิต 6	Data	D6	เอาต์พุท
9	Data บิต 7	Data	D7	เอาต์พุท
10	Ackowledge	Status	S6	อินพุท
11	$\overline{\text{Busy}}$	Status	S7	อินพุท
12	$\overline{\text{Out of paper}}$	Status	S5	อินพุท
13	Select	Status	S4	อินพุท
14	Line Feed	Control	C1	เอาต์พุท
15	Error	Status	S3	อินพุท
16	Initial	Control	C2	เอาต์พุท
17	$\overline{\text{Select In}}$	Control	C3	เอาต์พุท
18-25	Ground			

ตารางที่ 5.2 หน้าที่และการทำงานของขาต่างๆของพอร์ตขนาน

พอร์ตขนาน	แอดเดรสของ	แอดเดรสของ	แอดเดรสของ
	DATA	STATUS	CONTROL
LPT1	&H378	&H379	&H37A
LPT2	&H3BC	&H3BD	&H3BE
LPT3	&H278	&H279	&H27A

ตารางที่ 5.3 แสดงแอดเดรสของรีจิสเตอร์ของพอร์ตขนาน

5.3.1 พอร์ตเอาต์พุต

พอร์ตที่ทำหน้าที่เป็นพอร์ตเอาต์พุตคือ พอร์ต DATA และพอร์ต CONTROL สำหรับพอร์ต DATA ตำแหน่งบิท D0-D7 สามารถใช้งานเป็นเอาต์พุต ได้ทั้งหมด ส่วนพอร์ต CONTROL มีบิทที่ทำหน้าที่เป็นเอาต์พุตเพียง 4 บิท คือ C0-C3 ส่วนตำแหน่งอื่นๆไม่ได้ใช้งานหรือถูกสงวนไว้ใช้กับงานอื่น

การเขียนข้อมูลไปยังรีจิสเตอร์ DATA นั้นข้อมูลที่ส่งออกไปกับสถานะลอจิกที่ขาพอร์ต DATA นั้นข้อมูลที่ส่งออกไปกับสถานะลอจิกที่ขาพอร์ต DATA จะตรงกัน แต่สำหรับพอร์ต CONTROL นั้น บิท C0,C1 และ C3 จะกลับสถานะ โดยเมื่อป้อนค่าหนึ่งค่าใดไปยังรีจิสเตอร์ CONTROL ที่พอร์ต CONTROL จะทำการเปลี่ยนแปลงข้อมูลที่อยู่ในบิท C0.C1.C3 ให้มีสถานะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตรงกันข้าม ดังนั้นเมื่อต้องการส่งข้อมูลออกไปยังพอร์ต CONTROL จะต้องมีการกลับค่าของข้อมูลในบิท C0,C1,C3 ก่อนเสมอ

สำหรับโปรแกรมที่ใช้ในการเขียนข้อมูลไปยังพอร์ตขนานนั้น สามารถเขียนได้โดยง่าย โดยใช้ภาษาระดับต่ำเช่น ภาษาแอสเซมบลี้ ไปจนถึงภาษาระดับสูงเช่น ภาษาเบสิก, เทอร์โบปาสคาล หรือ เทอร์โบซี สำหรับตัวอย่างการส่งข้อมูลออกไปยังพอร์ตเอาต์พุตของพอร์ตขนานด้วยการเขียนโปรแกรมภาษาแอสเซมบลี้มีดังนี้

โปรแกรมภาษาแอสเซมบลี้

```
mov dx,378h
mov al,value
out  dx,al
```

5.3.2 พอร์ตอินพุต

พอร์ตที่ทำหน้าที่เป็นพอร์ตอินพุตคือ พอร์ต STATUS มีบิตที่ใช้งานเป็นอินพุตเพื่อรับข้อมูลจากภายนอก 5 บิท คือ S3-S7 โดยบิท S7 นั้นมีการกลับสถานะอยู่ ดังนั้นการอ่านค่าข้อมูลในบิท 7 จะได้ค่าออกมาเป็นค่าตรงกันข้าม ตำแหน่งแอดเดรสของพอร์ต STATUS จะอยู่ถัดจากแอดเดรสของพอร์ต DATA หนึ่งตำแหน่ง ดังนั้นเพื่อเป็นการง่ายต่อการทำความเข้าใจ การเขียนโปรแกรมไปยังพอรต STATUS จึงนิยมระบุตำแหน่งแอดเดรสเป็น BaseAddr+1 (ปกติมีค่าเท่ากับ &H378+1) สำหรับตัวอย่างการอ่านข้อมูลจากพอร์ตอินพุตด้วยการเขียนโปรแกรมภาษาแอสเซมบลี้ มีดังนี้

โปรแกรมภาษาแอสเซมบลี้

```
mov dx,378H
in  al,dx
```

5.3.3 พอร์ตสำหรับคอมพิวเตอร์รุ่นใหม่

สำหรับในคอมพิวเตอร์รุ่นใหม่ๆนั้น พอร์ต DATA สามารถใช้เป็นอินพุตได้ด้วย โดยจะมีบิทที่ทำหน้าที่กำหนดทิศทางของข้อมูลอยู่ที่พอร์ต CONTROL บิท 5 (C5) ถ้าบิทนี้เป็น 0 จะเป็นการกำหนดให้พอร์ตนี้เป็นเอาต์พุตถ้าบิทนี้เซตเป็น 1 จะเป็นการดิสเอเบิ้ลเอาต์พุต ทำให้สามารถอ่านสถานะลอจิกจากภายนอกได้ ซึ่งก็คือการทำหน้าที่เป็นอินพุตนั่นเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6
อุปกรณ์ต่างๆและวัสดุในการทดลอง

6.1 ชิ้นส่วนและอุปกรณ์ต่างๆที่สำคัญ

1. Table ส่วนบน ซึ่งประกอบด้วย
   - -บอลล์สกรูตัวเล็ก
   - -บอลล์นัตตัวเล็ก
   - -อุปกรณ์จับยึดชิ้นงาน
2. Table ส่วนล่าง ซึ่งประกอบด้วย
   - -บอลล์สกรูตัวใหญ่
   - -บอลล์นัตตัวใหญ่
   - -Guide vane 1 คู่
3. กระบอกสูบลม 1 ลูก
4. โซลินอยด์วาล์ว 1 ชุด
5. ปั๊มลม
6. สายท่อลม
7. ชุดหัวPunch
8. ชุดDie
9. Stepping Motors 2 ตัว
10. สายพาน 2 เส้น
11. พูลเล่ย์สายพาน 2 ชุด
12. ฐานรองเป็นแผ่นเหล็กแผ่นเรียบ
13. คานจับยึดกระบอกสูบ
14. ชุดวงจร Drive motor
15. สายไฟที่ใช้ในการเชื่อมต่อวงจรและ Stepping Motor
16. เครื่องคอมพิวเตอร์
17. Power Supply

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.2 รายละเอียดต่างๆของชิ้นส่วนและอุปกรณ์ต่างๆโดยสังเขป

1. ในส่วนของ Table ส่วนบนนั้นจะมีชุดจับยึดชิ้นงานติดอยู่บนโครงเหล็กซึ่งจะเคลื่อนที่ไปตาม Guide vane โดยได้รับการส่งกำลังจาก Stepping Motor โดยผ่านทางสายพานมาขับบอลล์สกรู ซึ่งในการหมุนแต่ละรอบของบอลล์สกรูนั้นจะทำหน้าที่ส่งกำลังไปทำหน้าที่ในการขับโครงเหล็กซึ่งยึดกับชุดจับยึดชิ้นงานอีกทีหนึ่ง ทั้งนี้ส่วนจับยึดมีลักษณะเป็นเหล็กแผ่นรูปตัว C ประกบกัน ซึ่งสามารถจับยึดชิ้นงานได้ขนาดกว้าง 10 เซนติเมตร ยาว 15 เซนติเมตร โดยการเคลื่อนที่ของTableส่วนบนนั้นจะเป็นการเคลื่อนที่ในแนวแกน x ได้ระยะทางทั้งสิ้น 18 เซนติเมตร
2. Table ส่วนล่างนั้นจะมี Guide vane 1 คู่ เป็นแท่งสเตนเลสกลมเป็นแกนนำให้ส่วนของ Table ส่วนล่างเลื่อนไปในแนวแกน y โดยได้รับการส่งกำลังจากStepping Motor ผ่านพูลเล่ย์สายพานมายังบอลล์สกรู ซึ่งการหมุนของบอลล์สกรูนั้นจะทำหน้าที่ในการขับ Table ให้เลื่อนไปในแนวแกน y ซึ่งเป็นระยะทางที่สามารถเลื่อนได้ทั้งสิ้น 23 เซนติเมตร
3. กระบอกสูบลม ใช้ทำหน้าที่ในส่งกำลังในการเจาะ (Punch) ชิ้นงาน โดยกระบอกสูบที่ใช้เป็นกระบอกสูบแบบสองทาง ไป-กลับ ด้วยลม ขนาดเส้นผ่านศูนย์กลางกระบอกสูบ 10 เซนติเมตร และระยะชัก 14 เซนติเมตร
4. โซลินอยด์วาล์ว ใช้ทำหน้าที่ในการควบคุมทิศทางการเลื่อนเข้าออกของก้านสูบเพื่อทำการเจาะชิ้นงาน โดยการปรับเปลี่ยนทิศทางของลมที่จะเข้ากระบอกสูบ ซึ่งจะได้รับสัญญาณจากคอมพิวเตอร์โดยผ่านพอร์ทเครื่องพิมพ์ ชุดขยายสัญญาณ มายังขดลวดในโซลินอยด์วาล์วเพื่อเหนี่ยวนำให้เกิดอำนาจแม่เหล็กเปลี่ยนทิศทางลมต่อไป
5. ปั๊มลม ใช้ในการให้กำลังในการขับเคลื่อนกระบอกสูบ โดยปั๊มลมที่ใช้นี้เป็นปั๊มลมขนาด 10 กิโลกรัมต่อตารางเซนติเมตร
6. สายลม ทำหน้าที่ในการส่งลมจากปั๊มลมมายังโซลินอยด์วาล์ว และจากโซลินอยด์วาล์วไปยังกระบอกสูบ ซึ่งใช้ขนาดเส้นผ่านศูนย์กลาง 0.5 เซนติเมตร
7. ชุดหัวเจาะจะทำหน้าที่ในการเจาะชิ้นงาน โดยหัวเจาะจะมีการออกแบบให้มีการเจาะเป็นแบบ 2จังหวะ โดยในจังหวะแรกจะเป็นการจับยึดชิ้นงานให้ติดแน่นกับตัว Die ก่อน จากนั้นในจังหวะที่ 2 ตัวหัวเจาะจึงจะทำการเจาะชิ้นงาน ทั้งนี้หัวจับยึดในจังหวะแรกทำจากอลูมิเนียมเพื่อให้มีน้ำหนักเบาเพื่อป้องกันการยืดของสปริงที่รับน้ำหนักของตัวมันอยู่ ในส่วนของหัวเจาะในจังหวะที่ 2 ทำจากเหล็กคาร์บอน ชุบแข็งที่ 880 องศาเซลเซียส Tempering ในน้ำมัน เพื่อให้มีความแข็งสูงเพราะต้องใช้ในการเจาะชิ้นงานโดยตรง
8. ชุด Die ทำหน้าที่ในการรองรับการกดเจาะชิ้นงานโดยตรง ซึ่งบริเวณกดเฉือนรอบๆหัวเจาะจะต้องพอดีกับรู Die และรู Die จะต้องมีความแข็งแรงมาก จึงต้องทำจากเหล็กชุบแข็ง โดยจะทำการชุบแข็งที่อุณหภูมิ 880 องศาเซลเซียส Tempering ในน้ำมัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

9. Stepping Motor ที่เลือกใช้ในโครงงานนี้ เราจะเลือกใช้ชนิดแบบ 6 สาย 4 เฟส หรือเรียกว่าชนิดยูนิโพลาร์ โดยส่วนประกอบที่สำคัญของมอเตอร์ชนิดนี้สามารถสังเกตได้โดยง่าย เนื่องจากจะมีสายไฟทั้งหมด 6 เส้นด้วยกัน และในการพันขดลวดนั้นจะมีการพันทั้งหมด 2 ขด บนขั้วแม่เหล็กของสเตเตอร์ ซึ่งในแต่ละขดจะทำให้เกิดขั้วแม่เหล็กในทิศทางตรงข้ามกัน
10. สายพานที่ใช้จะเป็นสายพานแบบมีฟันโดยมีขนาดเท่ากับขนาดฟันของพูลเล่ย์ และความยาวเส้นรอบวงคือ 20 และ 33 เซนติเมตร
11. พูลเล่ย์สายพานที่ใช้จะมีขนาดของฟันเท่ากับขนาดฟันของสายพาน โดยเส้นผ่านศูนย์กลางเท่ากับ 2 และ 4 เซนติเมตร
12. ฐานรองเป็นแผ่นเหล็กแผ่นเหล็กแผ่นเรียบ พร้อมกับขาตั้งที่ทำการปรับระดับไว้เรียบร้อยแล้วเพื่อช่วยให้การเคลื่อนที่ของ Table มีความถูกต้องมากยิ่งขึ้น โดยในการปรับระดับขาตั้งจะช่วยให้ Table สามารถเคลื่อนที่ได้เป็นระนาบที่ไม่มีการเอียงไปในทิศทางใดทิศทางหนึ่ง ซึ่งจะช่วยส่งผลให้การเจาะชิ้นงานนั้นถูกต้องมากยิ่งขึ้น ขนาดโดยประมาณของฐานรอง กว้าง 80 เซนติเมตร ยาว90 เซนติเมตร
13. คานจับยึดกระบอกสูบจะใช้เหล็กรูปตัว U ตั้งขึ้นไปในแนวดิ่ง และใช้เหล็กฉากวางเพื่อรองรับกระบอกสูบในแนวนอน ทั้งนี้การใช้เหล็กฉากก็เพื่อรองรับ load ในแนวดิ่ง โดยป้องกันการเกิด Bending moment และง่ายต่อการจับยึด
14. วงจร Drive motor จะประกอบด้วย ทรานซิสเตอร์ TIP 41 จำนวน 8 ตัว
15. สายไฟที่ใช้ในการเชื่อมต่อวงจรและ Stepping Motor จะใช้สายแพในการเชื่อมต่อจากชุดควบคุมมายัง Hardware และจาก พอร์ต Printer มายังชุดควบคุม และใช้สายเดี่ยวในการเชื่อมต่อภายในวงจรควบคุม
16. เครื่อง Computer ที่ใช้นั้นควรจะเป็นเครื่องที่มี spec เป็น Pentium ขึ้นไปเพื่อการประมวลผลที่รวดเร็ว และเพื่อการแสดงผลที่มีประสิทธิภาพ แต่ถ้าใช้รุ่นต่ำกว่านี้ก็ไม่เป็นไร โดยมีโปรแกรมที่จำเป็นสำหรับการควบคุมการทำงาน คือ Delphi
17. Power Supply ที่ใช้นั้นควรจะเป็นชนิดที่สามารถปรับปริมาณกระแสและความต่างศักย์ได้ เพื่อการใช้งานที่เหมาะสมของขนาดของ Stepping Motor และขนาดหรือน้ำหนักของ Table

6.3 วัสดุในการทดลอง

จากการทดลองใช้เหล็กเหนียวในการทดลองซึ่งมีรายละเอียดดังนี้

-เหล็กแผ่นเรียบและเหล็กแผ่นบาง

-เหล็กแผ่นเรียบผลิตจากเหล็กไม่ผสมที่มีความหนาน้อยกว่า 0.5 มิลลิเมตร เหล็กแผ่นขาวเป็นเหล็กแผ่นเรียบ โดยผ่านการชุบไฟฟ้า หรือเคลือบดีบุก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สัญลักษณ์สำหรับเหล็กแผ่นชนิดต่างๆ

ตามขอบเขตความแข็ง	โค๊ดตัวอักษร	A	B	C	D
	เขตความแข็ง HR 30 T	48...56	54...61	57...65	66...73

ตารางที่ 6.1 แสดงสัญลักษณ์ค่าความแข็งสำหรับเหล็กแผ่นชนิดต่างๆ

ตามผิวที่สำเร็จ	การคัด	สัญลักษณ์	ลักษณะของผิว
แผ่นเรียบ	คัดครั้งที่1	W1	ผิวใช้ทาสีหรือใช้พิมพ์ได้
	คัดครั้งที่2	W2	ผิวและรูปทรงไม่ดี
เหล็กแผ่นชุบสังกะสี	คัดครั้งที่1	W1	ผิวใช้ทาสีหรือใช้พิมพ์ได้
	คัดครั้งที่2	W2	ผิวมีตำหนิเล็กน้อย
	คัดครั้งที่3	W3	ผิวมีตำหนิมาก
เหล็กแผ่นชุบสังกะสี	มาตรฐาน	W1	ผิวไม่มีตำหนิ
ด้วยวิธี Electrolyte	ปานกลาง	W2	แผ่นเหล็กคุณภาพต่ำที่คัดออกจากมาตรฐาน

ตารางที่ 6.2 แสดงสัญลักษณ์ และ ลักษณะผิวของเหล็กแผ่นชนิดต่างๆ

เหล็กแผ่นขาวตามกรรมวิธีชุบดีบุก

ชนิดของการเคลือบดีบุก	ชุบด้วยความร้อน		เคลือบดีบุกโดยการ Electrolyte							เคลือบด้านเดียว	
			ดีบุกชนิดเดียวกันทั้งสองด้าน				ดีบุกต่างชนิดกัน				
สัญลักษณ์ชนิดของดีบุก	F24	F30	E1	E2	E3	E4	E3_1	E4_1	E4_2	E1_0	E2_0
นน.ที่เคลือบ / ตร.ม.	24	30	5,6	11,2	16,8	22,4	16,8/5,6	22,4/5,6	22,4/11,2	5,6/0	11,2/0

ตารางที่ 6.3 แสดงสัญลักษณ์ชนิดของดีบุก และ นน.ที่เคลือบ/ตารางเมตร

-สัญลักษณ์ของเหล็กแผ่นเรียบความแข็งอยู่ในช่อง A คัดครั้งที่ 1 (W1) แผ่นขนาด 1000 มม. x 850 มม.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

-สัญลักษณ์ เหล็กแผ่นขาวเคลือบดีบุก โดยวิธีชุบไฟฟ้า ความแข็งอยู่ในช่วง B เคลือบดีบุกชนิดเดียวกันทั้ง 2 ด้าน คือ มาตรฐาน (W1) ขนาด 1000 มม. x 850 มม. แผ่นขาว BE 2 – W1 1000 x 850

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 7
โปรแกรมคอมพิวเตอร์

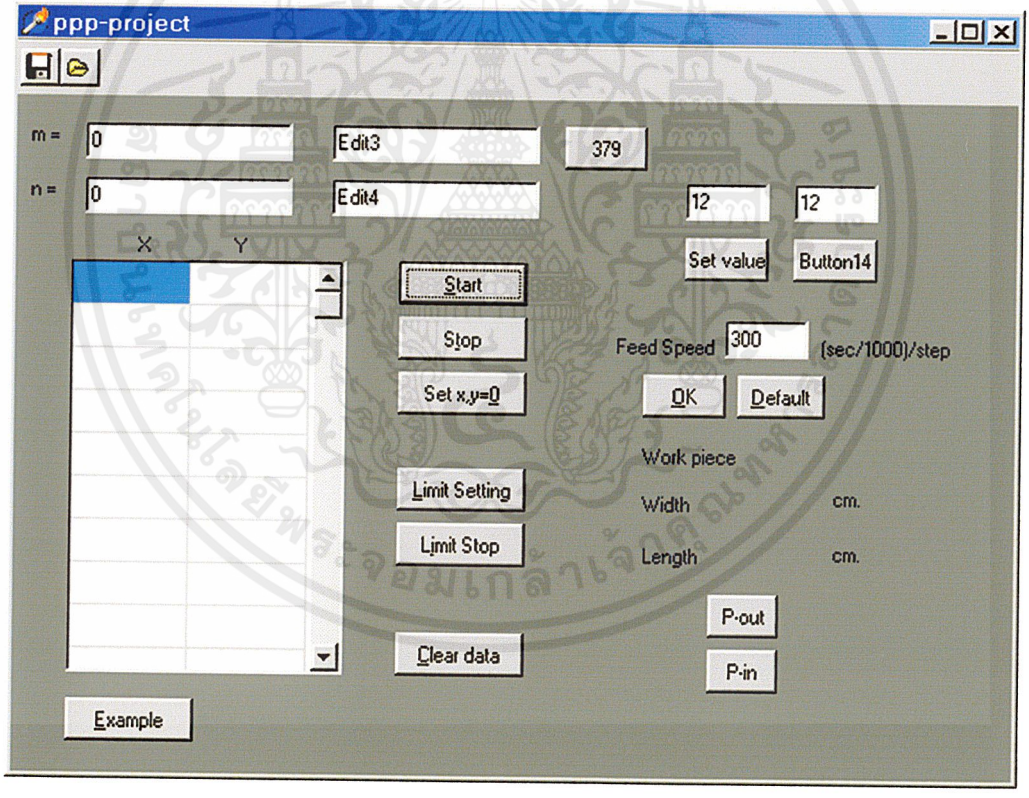
7.1 ส่วนประกอบของโปรแกรม

โปรแกรมที่เขียนด้วย Delphi จะแบ่งส่วนสำคัญออกเป็น 2 ส่วน คือ

1. Form
2. Source code

7.1.1 Form เป็นวินโดว์อันหนึ่งมีคุณสมบัติเหมือนวินโดว์ทั่วไป คือ ประกอบด้วย คอนโทรล เมนูปุ่มมินิไมซ์ ปุ่มแมกซิไมซ์ ไตเติลบาร์ และขอบหน้าต่างที่สามารถปรับขนาดได้

Form ในโปรเจคนี้ มีหน้าควบคุมซึ่งมีลักษณะดังรูป



รูปที่ 7.1 Control Parnel ของโปรแกรม

<u>รายละเอียดส่วนต่างๆ</u>

1. ช่องแสดงผล m,n - แสดงตำแหน่งเฟสของสเต็ปปิ้งมอเตอร์ที่ถูกกระตุ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. ช่อง X,Y - ช่องป้อนตำแหน่งจุดที่ต้องการ *punch*
3. ปุ่ม Example - แสดงตัวอย่างตำแหน่งของจุดที่ punch
4. ช่อง Edit 3 - แสดงค่าต่างๆในการนับสเต็ป
5. ช่อง Edit 4 - แสดงสถานะการทำงานของเครื่อง เช่น piston out , piston back เป็นต้น
6. ปุ่ม 379 - ปุ่มกดเพื่อต้องการตรวจสอบค่าใน accumulator
7. ปุ่ม Start - ปุ่มเริ่มการทำงาน
8. ปุ่ม Stop - ปุ่มหยุดการทำงานเมื่อสเต็ปปิ้งมอเตอร์หยุดการเคลื่อนที่
9. Set X,Y=0 - ปุ่มset ตำแหน่งของ table ทั้งแกน X และ แกน Y ให้กลับมาที่จุดเริ่มต้นตอนทำงานครั้งแรก
10. Limit setting - ปุ่ม set ตำแหน่งของ table ทั้งแกน X และแกน Y ให้อยู่ตำแหน่ง (0,0)
11. Limit stop - ปุ่มหยุดการ set ตำแหน่ง
12. Clear data - ปุ่มลบข้อมูลที่ป้อนค่าไว้
13. ปุ่ม Set value และปุ่ม buttom14 - ปุ่ม set ขนาดของชิ้นงาน (maximun=15*10)
14. ช่อง Feedspeed - ป้อนค่าความเร็วของสเต็ปปิ้งมอเตอร์ (*sec/1000 per step*)
15. ปุ่ม OK - ปุ่มยืนยันค่าความเร็วรอบที่ป้อน
16. ปุ่ม Default - ปุ่มยกเลิกค่าความเร็วที่ป้อน และกลับไปยังค่าความเร็วเดิม (300 sec/1000 per step)
17. P-out - ปุ่มทดสอบ piston out (punch ชิ้นงาน)
18. P-in - ปุ่มทดสอบ piston in (punch ชิ้นงาน)

7.1.2 Source code เป็นตัวโปรแกรมหลักที่ใช้ควบคุมการทำงานต่างๆ โดยภาษาที่ใช้ในโปรแกรม Delphi คือ ภาษา Pascal โครงสร้างของ code-editor เป็นดังนี้

```
Unit Unit1; Unit ชื่อของ unit
Interface
Uses
Windows,Messages,SysUtils,Clesses,Graphics,Controls,Forms,Dialogs;
การเรียกใช้คอมโพเนนท์ต่างๆ
type
Tform1=class(Tform) ชื่อของ form = class(tfrom) และสามารถประกาศชนิดของตัวแปรขึ้นมาใหม่ได้
Private
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

{ Private declarations } การประกาศ procedure หรือ function ที่ใช้ได้เฉพาะ unit นี้เท่านั้น

{ Public declarations } การประกาศ procedure หรือ function ที่สามารถเรียกใช้ได้จาก unit อื่นได้

end;

var

Form1:Tform1; ชื่อของ form:Tform1;และตัวแปรอื่นๆ

Implementation หลังจาก Implementation จนถึง end. เป็นตัวโปรแกรมที่ควบคุมการทำงาน

{ $R*.DFM } เป็นการ link กับ form ห้ามลบเด็ดขาด

เป็นการควบคุมที่มี procedure และ function ควบคุมการทำงานต่างๆ

end.

7.2 ขั้นตอนการใช้งาน

เริ่มต้นต้อง set หาจุดอ้างอิงของเครื่อง (พิกัด (0,0)) โดยการกดปุ่ม Limit setting ต่อจากนั้นก็ทำการกำหนดค่าพิกัด X,Y ที่ต้องการ punch เมื่อต้องการดูตัวอย่างชิ้นงานก่อน punch โดยกดปุ่ม Example ต่อมาก็กำหนดค่าความเร็วที่ต้องการ จากนั้นก็กดปุ่ม Start เริ่มการทำงาน ซึ่งจะสามารถสังเกตสถานะต่างๆได้

7.3 หลักการเขียนโปรแกรมควบคุมการเคลื่อนที่ของหัวเจาะ

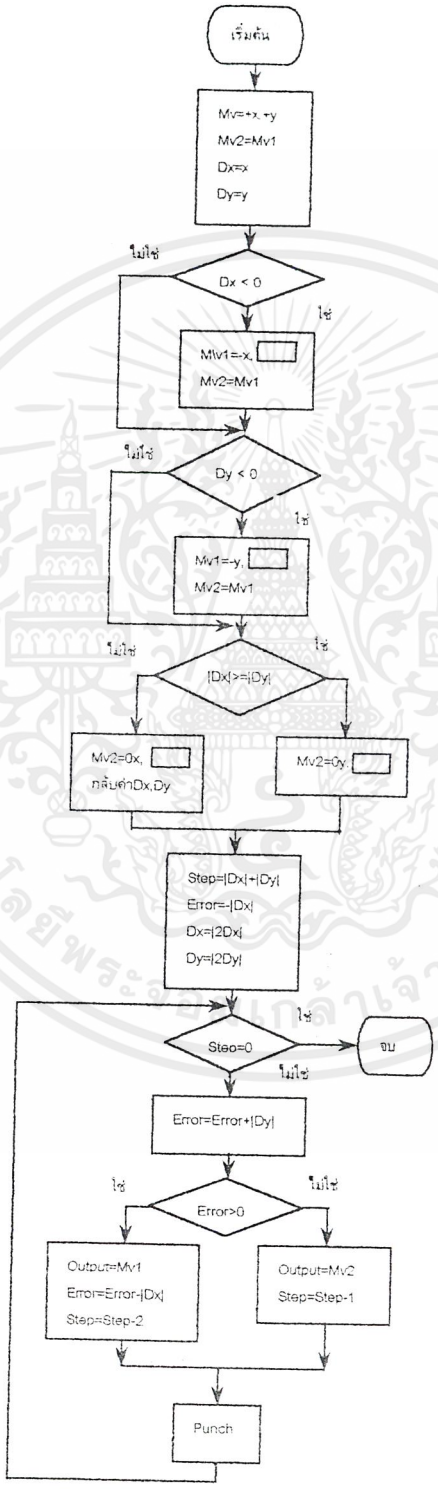
ในการหมุนของมอเตอร์เพื่อเคลื่อนที่หัวเจาะไปยังจุดต่างๆ จะเป็นไปตามสเต็ป ถ้าค่าของตำแหน่ง X และ Y มีค่าเท่ากัน ก็จะขับมอเตอร์ X และ Y ไปพร้อมกันเมื่อลากเส้นจะได้เส้นเฉียง 45 องศา ถ้าค่า X และ Y มีค่าเป็น 0 วงจรก็จะขับมอเตอร์ตัวที่เหลือให้ทำงานลากเส้นไปทิศทางของแกนที่เหลือ แต่ถ้าค่า X และ Y มีค่าไม่เท่ากัน และไม่มีค่าเป็น 0 จะทำอย่างไรจึงจะสามารถขับมอเตอร์ให้ลากเส้นได้ตรงที่สุด

มีวิธีการหนึ่งที่ใช้ได้ดีในการควบคุมการหมุนของสเต็ปปิ้งมอเตอร์คือ การใช้อัลกอริทึมของ Bresenham หลักการของอัลกอริทึมแบบนี้ก็คือ หาค่าผิดพลาดในตัวแปร(error) ซึ่งเกิดจากการเปรียบเทียบค่าในอุดมคติของแต่ละจุดกับค่าจริง ค่าผิดพลาดนี้จะเป็นข้อจำกัดทางฮาร์ดแวร์ ดังนั้นจุดที่แท้จริงในการลากเส้นคือ ค่าประมาณที่ใกล้เคียงที่สุด

รูปที่ 7.2 เป็นโฟลว์ชาร์ตของซอฟต์แวร์ที่ใช้ควบคุมการลากเส้นของเครื่องเจาะ กำหนดเริ่มต้นเป็น (X1,Y1) และจุดสุดท้ายเป็น (X2,Y2) ให้ dX = X2-X1 และ dY = Y2-Y1 จากนั้นดูค่าของ dX และ dY เป็นลบหรือบวก ขั้นแรกกำหนดค่า MV1 และ MV2 จากนั้นทำค่า dX,dY ให้เป็นบวก ถ้าหากค่าเดิมเป็นลบแล้วนำมาเปรียบเทียบกันระหว่าง dX กับ dY แล้วกำหนดค่าของ MV2 ต่อจาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

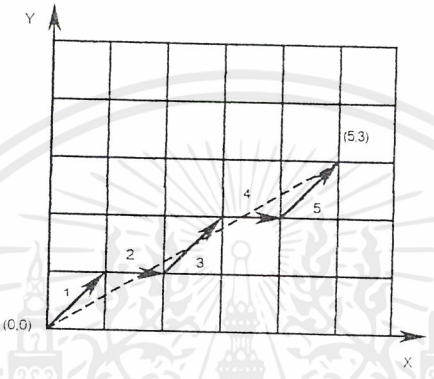
นั้นก็หาลักษณะการเดินทีละสเต็ปด้วยวิธีการใช้ค่า error ส่งค่าเอาท์พุทออกไปทางมอเตอร์ด้วยค่าของ MV1 หรือ MV2 สุดท้ายลดค่าสเต็ปลงจนเป็นศูนย์ก็จะได้เส้นที่เหมือนตรงที่สุด ตารางที่ 7.1 แสดงลักษณะการเคลื่อนที่ไปยังตำแหน่งต่างๆของเครื่องเจาะ



รูปที่ 7.2 โฟลว์ชาร์ตของซอฟต์แวร์ที่ใช้ควบคุมการเคลื่อนที่หัวเครื่องเจาะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วิธีการเคลื่อนที่ของหัวเจาะจะเป็นลักษณะนี้ทั้งหมด ไม่ว่าจะเคลื่อนที่เป็นรูปวงกลม เส้นโค้งหรือเส้นลักษณะใดก็ตามจากตัวอย่างการเคลื่อนที่ของหัวเจาะเป็นทแยงมุมในรูปที่ 7.3 จากรูปพบว่า dX = 5 ,dY = 3 ซึ่งเป็นบวกทั้งคู่ ดังนั้น MV1 = +X,0Y เมื่อได้ค่าของ MV1 และ MV2 แล้ว ก็หาวิธีการเคลื่อนที่เป็นสเต็ปด้วยการคำนวณตามโฟลว์ชาร์ตรูปที่ 7.2 ผลการคำนวณดูได้จากตารางที่ 7.2



รูปที่ 7.3 การเขียนเส้นทแยงมุม 45 องศาจากจุด (0,0) ไปยัง (5,3)

dx	dy	Angle	MV1	dx>=dy		MV2
+	+	0-90	+X,+Y	Y	0-90	+X,0Y
				N	45-90	0X,+Y
-	+	90-180	-X,+Y	N	90-135	0X,+Y
				Y	135-180	-X,0Y
-	-	180-270	-X,-Y	Y	160-225	-X,0Y
				N	225-315	0X,-Y
+	-	270-360	+X,-Y	N	270-315	0X,-Y
				Y	315-360	+X,0Y

ตารางที่ 7.1 การเคลื่อนที่ไปยังตำแหน่งต่างๆของเครื่องเจาะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สเต็ป	ค่าสเต็ป	ค่า error	error > 0	เป็นค่าของ
1	8	1	ใช่	MV1
2	6	-3	ไม่ใช่	MV2
3	5	3	ใช่	MV1
4	3	-1	ไม่ใช่	MV2
5	2	5	ใช่	Mv1
-	0	-	-	-

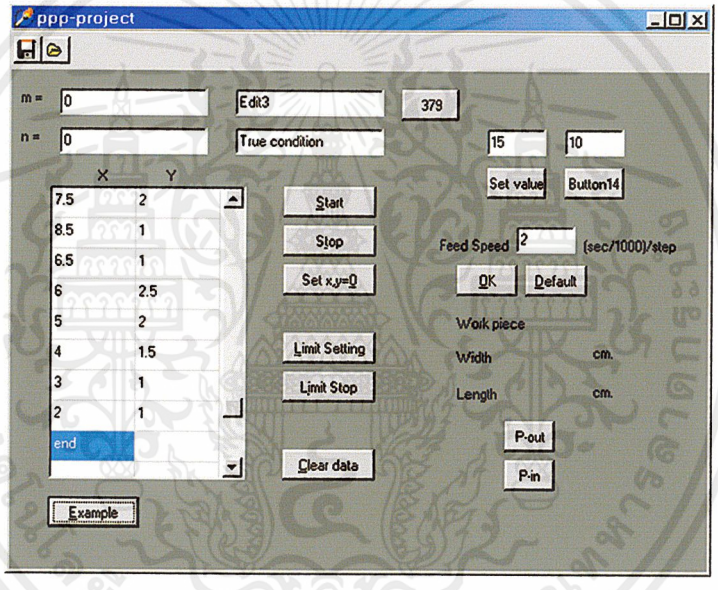
ตารางที่ 7.2 ผลการคำนวณค่าต่างๆตามโฟลว์ชาร์ต เพื่อกำหนดตำแหน่งของเครื่องเจาะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 8
ขั้นตอน และ ผลการทดลอง

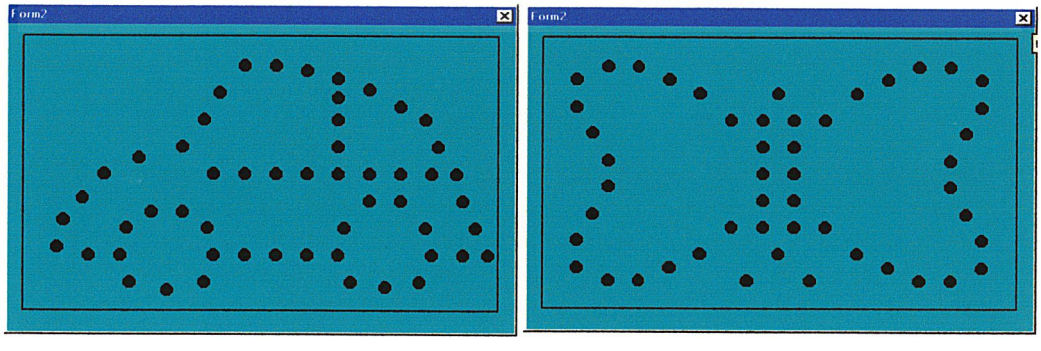
8.1 ขั้นตอนการทดลอง

1. ตัดชิ้นงานขนาด 170x200 มิลลิเมตร
2. นำชิ้นงานสอดเข้ากับที่จับยึด
3. เปิดเครื่อง และ รันโปรแกรม (ที่แนบอยู่ด้านหลัง)
4. ป้อนค่าจุดต่างๆที่ต้องการ punch ดังแสดงในรูปที่ 8.1



รูปที่ 8.1 แสดงการป้อนค่าที่ต้องการ punch

5. กดปุ่ม Example เพื่อดูชิ้นงานก่อนการเจาะจริง ดังรูปที่ 8.2



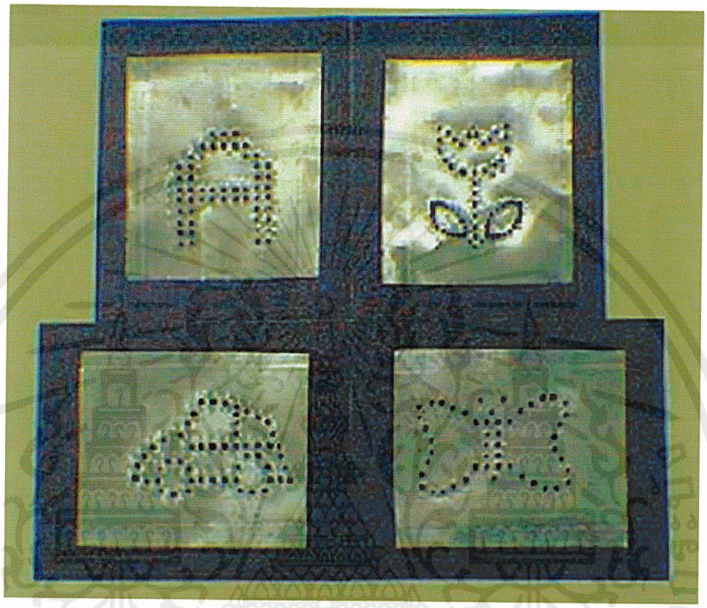
รูปที่ 8.2 แสดงตัวอย่างชิ้นงานก่อนการเจาะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. กำหนดความเร็วของมอเตอร์
7. กดปุ่ม start เริ่มการเจาะชิ้นงาน

8.2 ผลการทดลอง

จากการทดลอง ชิ้นงานที่ได้มีลักษณะดังรูปที่ 8.3



รูปที่ 8.3 แสดงตัวอย่างชิ้นงานที่ punch เสร็จเรียบร้อยแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 9
สรุปและแนวทาง

9.1 สรุป

1. การควบคุมของโครงงานนี้จะเป็นแบบอัตโนมัติ โดยที่การค้นหาจุดที่จะทำการเจาะชิ้นงานจะเป็นไปในลักษณะของการค้นหาแบบจุดต่อจุด ซึ่งเมื่อจะทำการค้นหาจุดต่อไปเพื่อที่จะทำการเจาะก็จะใช้จุดเดิมที่มีอยู่เป็นจุดอ้างอิง เป็นกระบวนการเช่นนี้เรื่อยไปจนกระทั่งครบทุกจุดตามต้องการ
2. การอ้างอิงแบบจุดต่อจุดนี้จะทำให้ได้ระยะทางที่สั้นที่สุดในการเคลื่อนที่ระหว่างจุดในแต่ละจุด ทำให้ใช้เวลาน้อยที่สุดในการทำงานในแต่ละรอบการทำงาน และ ในขณะเดียวกันการอ้างอิงแบบจุดต่อจุดก็มีข้อเสีย คือ เมื่อเกิดความผิดพลาดที่จุดใดจุดหนึ่งแล้ว ก็จะทำให้เกิดความคลาดเคลื่อนในจุดต่อไปเรื่อยๆ
3. ในส่วนของการเจาะชิ้นงานของหัวเจาะซึ่งทำจากเหล็กที่ไม่มีความแข็งเท่าที่ควร จึงทำให้ลักษณะของรอยเจาะไม่คมเท่าที่ควร และเมื่อทำการเจาะต่อไปนานๆเข้าก็จะทำให้เกิดการเจาะไม่เข้าในบางจุด และจากการที่หัวเจาะและ Die มีขนาดไม่ใกล้เคียงกันทำให้รอยการกดเจาะไม่คมเท่าที่ควรเนื่องจากการตัดเฉือนบริเวณขอบของหัวเจาะมีระยะให้ชิ้นงานยืดได้
4. ในส่วนของระบบนิวแมติกจะเกิดมีการรั่วของลมในบางจุด อาทิเช่น บริเวณข้อต่อระหว่างวาล์วกับสายลม บริเวณข้อต่อระหว่างกระบอกสูบกับสายลม เป็นต้น ทั้งนี้เนื่องจากเราใช้อุปกรณ์เก่าในการสร้าง เช่น กระบอกสูบลม วาล์ว เป็นต้น จึงทำให้เกิดความหลวมขึ้นในบางจุด และเนื่องจากเราใช้ปั๊มลมที่มีวาล์วควบคุมความดันทำให้ความดันจะถูกจำกัดไม่ให้ต่ำกว่าค่าหนึ่ง แต่ในขณะเดียวกันเมื่อมอเตอร์ปั๊มทำงานก็จะทำให้ความดันสูงกว่าค่าที่กำหนดนี้ จึงส่งผลให้การกดเจาะในแต่ละครั้งมีแรงไม่เท่ากัน ทำให้รอยการเจาะไม่สม่ำเสมอ
5. การเคลื่อนที่หาตำแหน่งโดยใช้ Stepping Motor ค่อนข้างที่จะเชื่อถือได้ เช่น ถ้าสั่งให้เดินทางไป 5 เซนติเมตร ก็จะเดินได้ใกล้เคียงมาก แต่สาเหตุที่การเคลื่อนที่ไม่ได้ 10 เซนติเมตรพอดี ก็เพราะว่าเป็นการควบคุมแบบ open loop ซึ่งจะไม่มีการป้อนค่ากลับมาตรวจสอบ จึงทำให้การเคลื่อนที่ไม่ได้ระยะทางตามที่กำหนดพอดี สาเหตุอีกประการหนึ่งก็คือ การไม่ได้ฉากกันระหว่าง Guide vane ของแกนทั้งสอง จึงทำให้แนวการเคลื่อนที่เบี่ยงเบนไป แต่ในขณะเดียวกันการใช้สายพานในการเชื่อมต่อระหว่างมอเตอร์กับบอลล์สกรู ซึ่งจะทำให้ลดปัญหาการเยื้องศูนย์ลงได้ แต่เมื่อใช้ไปนานๆสายพานอาจยืดได้ จึงควรมีการตรวจสอบและเปลี่ยนสายพานอยู่เสมอ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. ในส่วนของการจับยึดชิ้นงาน เนื่องจากเราใช้การจับยึดเป็นแผ่นเหล็กรูปตัว C จำนวน 2 แผ่นประกบกัน ทำให้แรงในการจับยึดไม่สม่ำเสมอทั่วทั้งแผ่นของชิ้นงาน เมื่อทำการกดเจาะชิ้นงานจึงทำให้คุณภาพของรอยเจาะไม่เท่ากันทุกรอย
7. จากการทดลองเราใช้ชิ้นงานที่มีขนาดบางเกินไปจึงทำให้ขณะทำการเจาะ ชิ้นงานเกิดการบิดงอไปเล็กน้อย ซึ่งส่งผลให้เกิดความคลาดเคลื่อนของตำแหน่งการเจาะขึ้นเล็กน้อย

9.2 แนวทางในการพัฒนา

1. การปรับปรุงให้รอยเจาะมีความคมและสม่ำเสมอ สามารถทำได้โดยการใช้เหล็กที่มีความแข็งสูงๆ แต่ในขณะเดียวกันก็ต้องไม่เปราะเกินไป และควรออกแบบให้ขนาดของหัวเจาะและ Die มีขนาดใกล้เคียงกันมากที่สุด
2. เพื่อป้องกันการเกิดความคลาดเคลื่อนในระบบนิวแมติก จึงควรใช้อุปกรณ์ที่มีคุณภาพดี และมีการซีลกันรั่วที่ดี
3. เพื่อป้องกันการเกิดความคลาดเคลื่อนในการค้นหาตำแหน่ง ควรมีการติดตั้งอุปกรณ์ป้อนค่ากลับเพิ่มเข้าไป เพื่อตรวจสอบว่าการเคลื่อนที่ค้นหาตำแหน่งได้ตรงตามจุดที่ต้องการแล้ว และควรมีการติดตั้ง Guide vane ของทั้งสองแนวแกนให้ตั้งฉากกัน
4. ควรมีการออกแบบระบบการจับยึดให้ออกแรงในการจับยึดได้สม่ำเสมอทั่วทั้งแผ่น ทั้งนี้ต้องสะดวกต่อการทำงานของเครื่องและการป้อนชิ้นงานด้วย
5. สามารถนำแนวคิดการเคลื่อนที่แบบสองแกนไปประยุกต์ใช้ในงานอื่นๆได้ เช่น ในเครื่อง Plotter เครื่องกัดขนาดเล็ก เป็นต้น
6. สามารถปรับปรุงให้การแสดงผลทางหน้าจอมีการแสดงผลว่า ขณะกำลังทำการเจาะที่จุดใด และจุดใดได้ทำการเจาะเสร็จสิ้นแล้วบ้าง
7. เพื่อความปลอดภัยควรมีการออกแบบและติดตั้งระบบความปลอดภัยในขณะเครื่องจักรกำลังทำงานด้วย
8. ควรมีการปรับขนาดของบอล์ลสกรูให้มีขนาดมุมของเกลียวเพิ่มมากขึ้นเพื่อให้การเคลื่อนที่หาตำแหน่งรวดเร็วขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

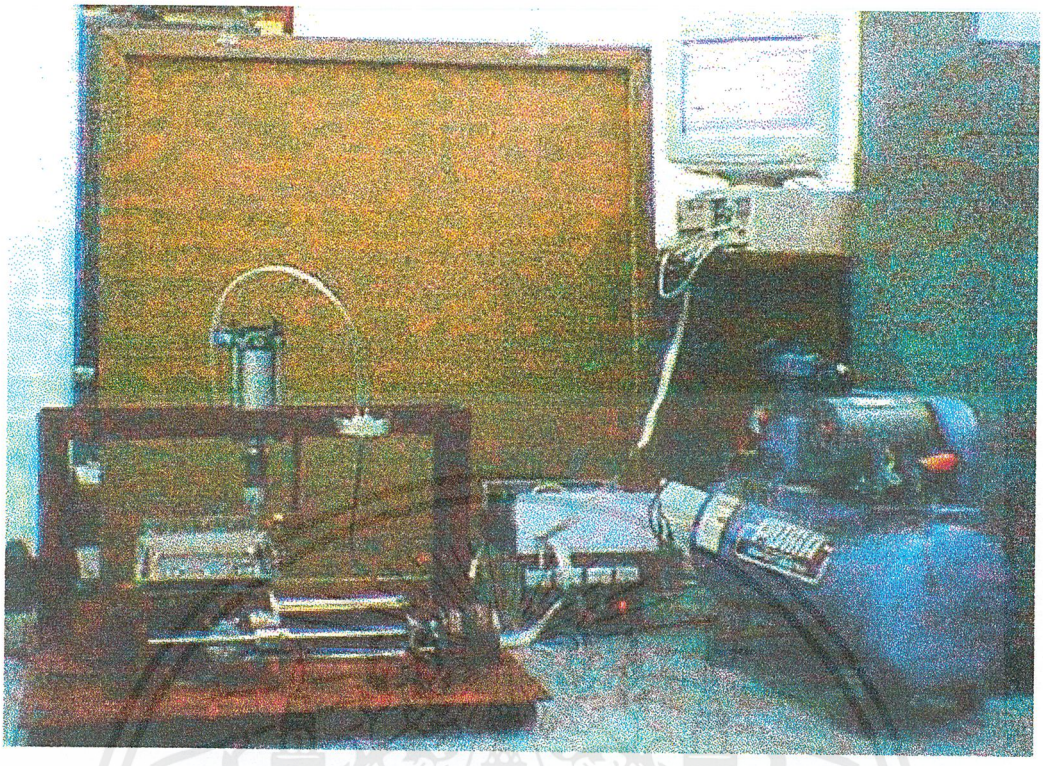
เอกสารอ้างอิง

1. เทคนิคการเชื่อมต่อ IBM PC , จิติ หนูแก้ว
2. เซมิคอนดักเตอร์ อิเล็กทรอนิกส์,ฉบับที่175
3. ไมโครเมาส์ , สมชาย อังศุโชติกุล , อดิศักดิ์ โหตระภวานนท์
4. IBM Microcomputer Architecture and Assembly Language , Norman S.Matloff
5. The IBM Personal Computer From the Inside Out , Murray Sargent and Richard L.Shoemaker
6. เครื่องกัดแนวดิ่ง CNC บนพื้นฐานไมโครคอมพิวเตอร์ , ทรงพล สิริรวมทรัพย์ , รังสรรค์ เสาวพุทธสุเวช
7. ตารางงานโลหะ , รศ.บรรเลง ศรนิล , ผศ.ประเสริฐ ก๊วยสมบรูณ์
8. Delphi 4 , กนก กุสุมาลย์นุกูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

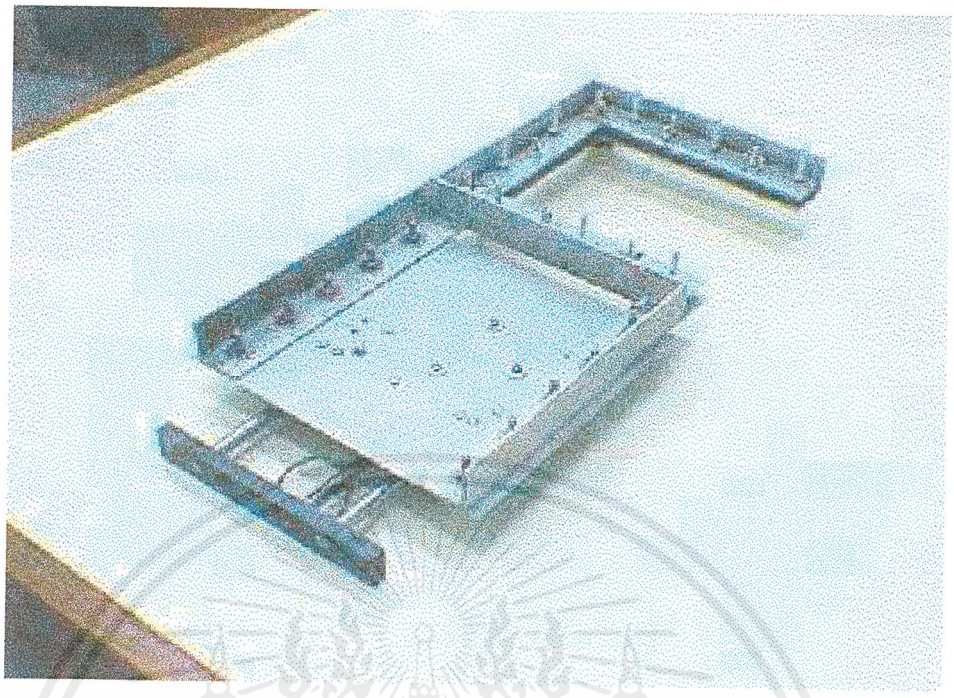
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



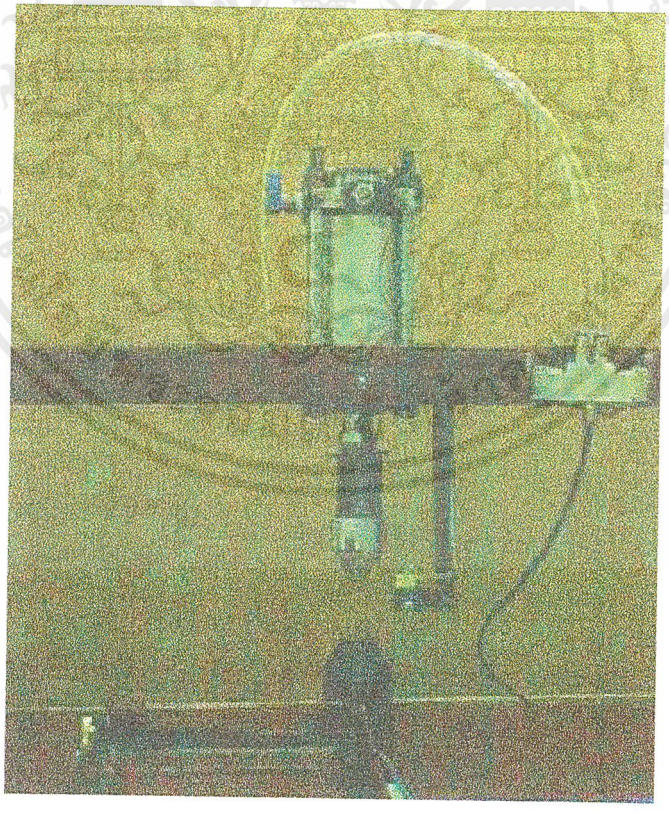
รูปที่ 1 แสดงภาพรวมแครื่องเจาะรูควบคุมด้วยคอมพิวเตอร์

รูปที่ 2 แสดงฐานล่างพร้อม Ball Screw และ Stepping Motor

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

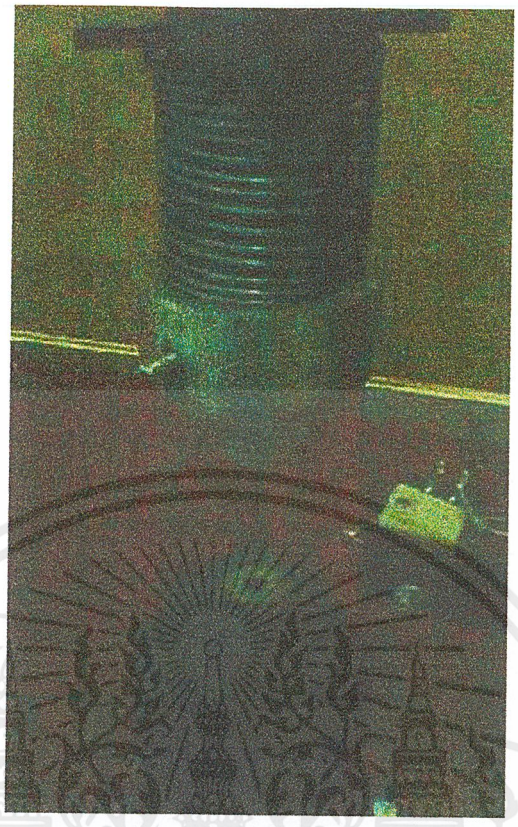


รูปที่ 3 แสดงฐานบนพร้อมที่จับยึด

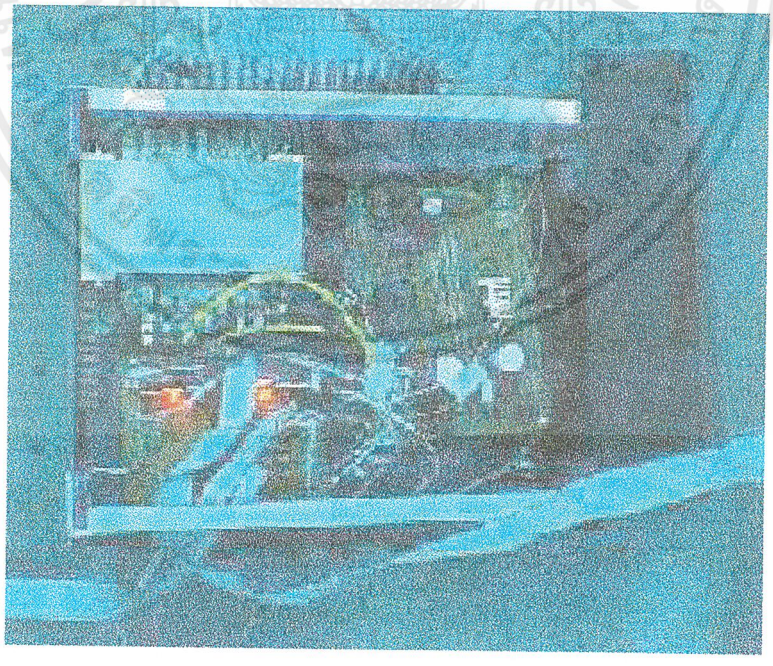


รูปที่ 4 แสดงระบบนิวแมติก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

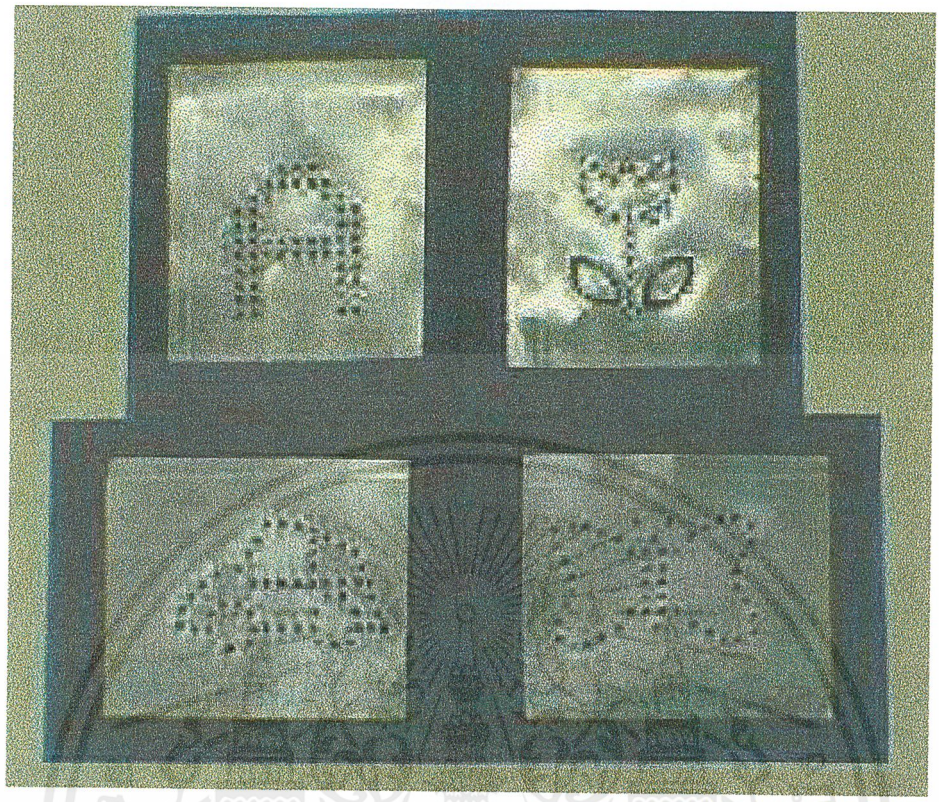


รูปที่ 5 แสดงหัวเจาะ และ Die



รูปที่ 6 แสดงวงจร Drive และ วงจรควบคุม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



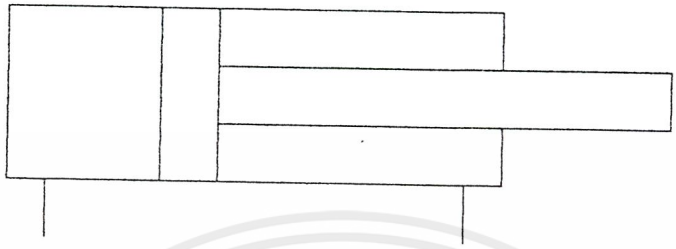
รูปที่ 7 แสดงชิ้นงานการทดลอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบนิวแมติก (Pneumatics systems)

ระบบนิวแมติก แบ่งออกได้เป็น 2 ส่วนใหญ่ๆ ดังนี้

1) กระบอกสูบ



-จากการศึกษาได้ เทนไซล์สเตร้ง ของสังกะสีเท่ากับ 20 กิโลกรัม/ตารางมิลลิเมตร

-กำหนดให้ขนาดรูที่จะทำการเจาะเท่ากับ 5 มิลลิเมตร ชิ้นงานหนา 1 มิลลิเมตร

-กำหนดความดันใช้งานเท่ากับ 0.03 กิโลกรัม/ตารางมิลลิเมตร

การคำนวณหาขนาดกระบอกสูบที่ต้องการ

จาก $\sigma = 0.6\tau$

จะได้ $\tau = 12$ กิโลกรัม/ตารางมิลลิเมตร

จาก พื้นที่รับแรงเฉือนในการเจาะรูเท่ากับ 15.7143 ตารางมิลลิเมตร

จาก $F = \tau A$ จะได้แรง $F = 188.5716$ กิโลกรัม

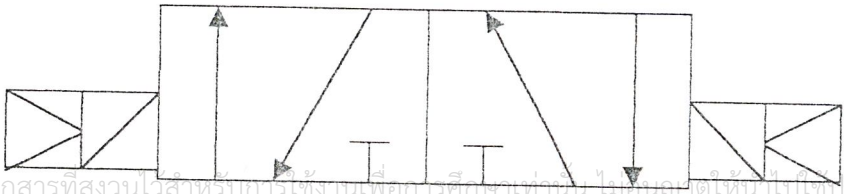
ที่กระบอกสูบ $F = PA$ ฉะนั้น พื้นที่ลูกสูบ $A = F/P$

จะได้พื้นที่ลูกสูบที่ต้องการเท่ากับ 6283.92 ตารางมิลลิเมตร เพราะฉะนั้นเส้นผ่านศูนย์กลางลูกสูบเท่ากับ 89.43 มิลลิเมตร

เพราะฉะนั้นจึงเลือกใช้กระบอกสูบดังนี้

- กระบอกสูบที่ใช้เป็นแบบ ไป-กลับ 2 ทาง
- ขนาดเส้นผ่านศูนย์กลางของกระบอกสูบมีขนาด 10 เซนติเมตร
- กระบอกสูบที่ใช้มีระยะชักประมาณ 14 เซนติเมตร

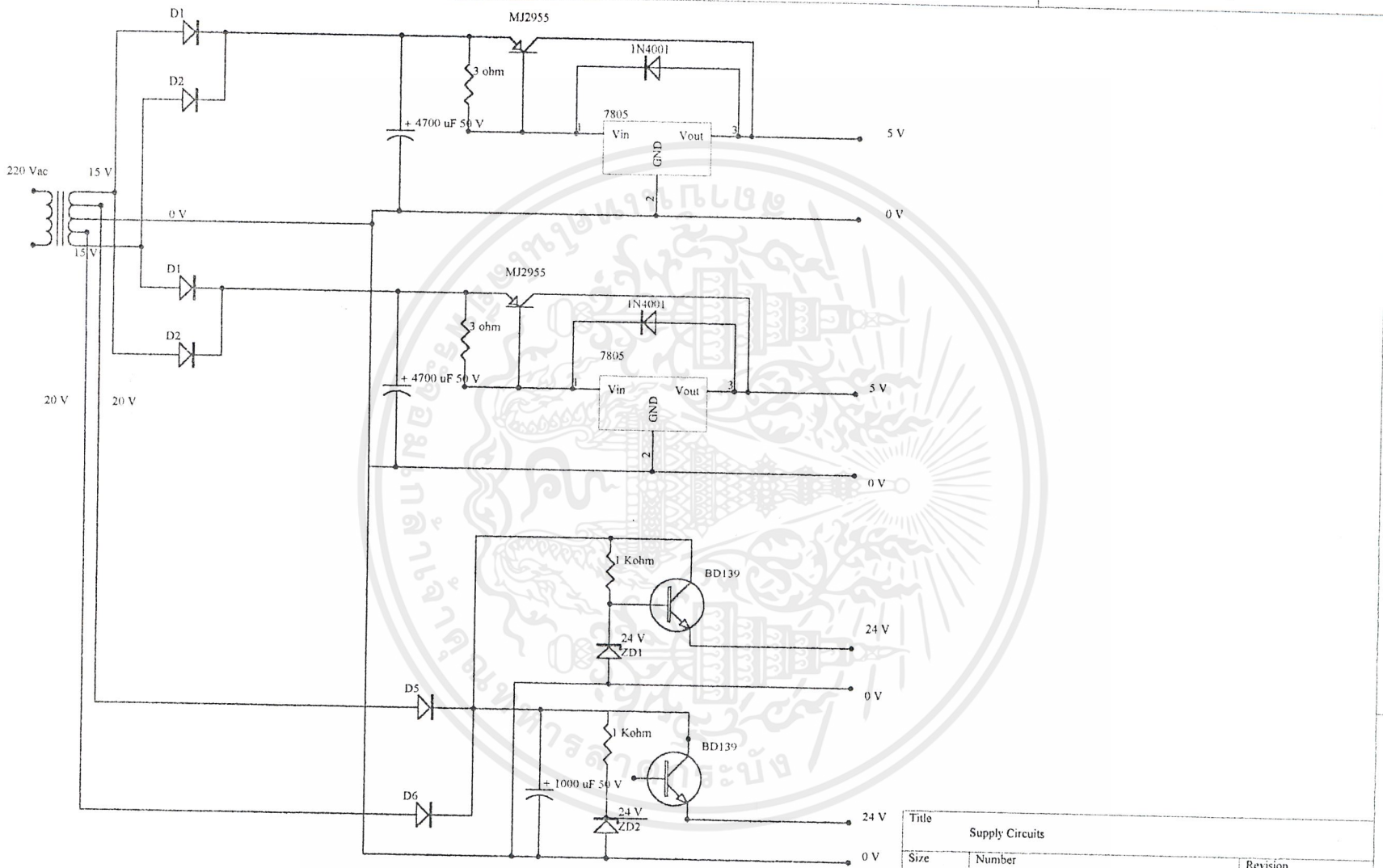
2) โซลินอยด์วาล์ว



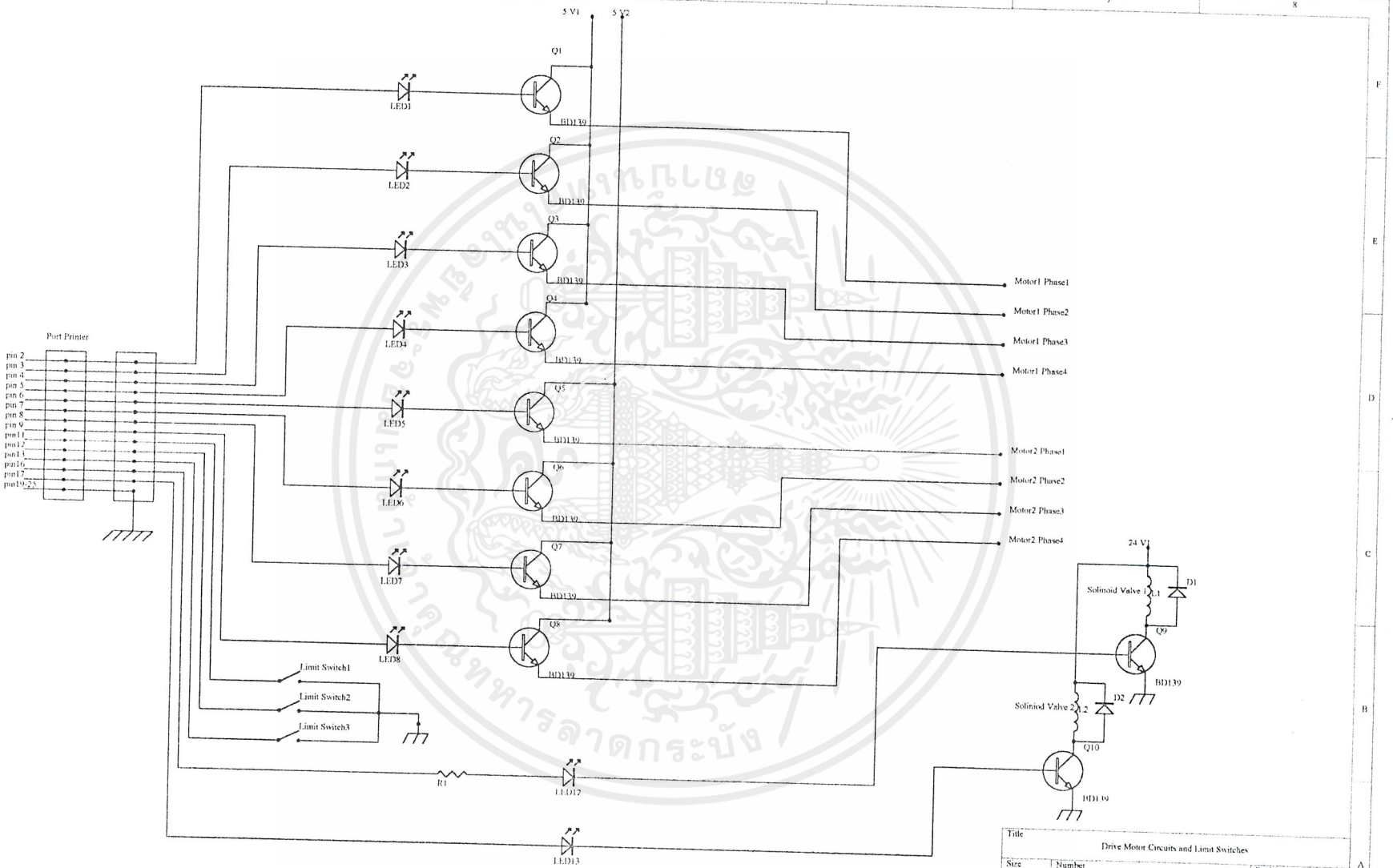
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

-ใช้โซลินอยด์วาล์ว ขนาด 24 V.DC. ซึ่งมีหลักการทำงาน คือ เมื่อมีสัญญาณ pulse จากคอมพิวเตอร์ผ่านวงจรขยายมายังขดลวดโซลินอยด์ ทำให้เกิดอำนาจแม่เหล็กจึงเกิดการเหนี่ยวนำให้กลไกภายในวาล์วทำการเปลี่ยนทิศทางลมให้เข้ายังห้องใดห้องหนึ่ง เช่น ถ้าเข้าห้องทางขวามือ (ตามรูป) ก็จะทำให้ลมออกไปทำให้ดันก้านสูบให้เลื่อนออก และลมจากกระบอกสูบระบายออกทางห้องซ้ายมือ ในทำนองเดียวกันถ้าเกิดการเหนี่ยวนำให้กลไกภายในวาล์วเลื่อนให้ลมเข้าทางห้องซ้ายมือ ก็จะทำให้ก้านสูบหดกลับ ในขณะที่ลมระบายออกทางห้องทางขวามือ เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Title	Supply Circuits		
Size A4	Number		Revision
Date:	22-Mar-2000	Sheet of	1/2
File:	A:\CIRCUIT1.SCH	Drawn By:	Panupat Tansiri


5 V1
5 V2
Q1
LED1
BD139
Q2
LED2
BD139
Q3
LED3
BD139
Q4
LED4
BD139
Q5
LED5
BD139
Q6
LED6
BD139
Q7
LED7
BD139
Q8
LED8
BD139
Port Printer
pin 2
pin 3
pin 4
pin 5
pin 6
pin 7
pin 8
pin 9
pin11
pin12
pin13
pin16
pin17
pin19-25
Limit Switch1
Limit Switch2
Limit Switch3
R1
LED12
LED13
Motor1 Phase1
Motor1 Phase2
Motor1 Phase3
Motor1 Phase4
Motor2 Phase1
Motor2 Phase2
Motor2 Phase3
Motor2 Phase4
24 V1
Solinoid Valve 1
L1
D1
Q9
BD139
Solinoid Valve 2
L2
D2
Q10
BD139
Title
Drive Motor Circuits and Limit Switches
Size
B
Number
Revision
Date
22-Mar-2000
Sheet of
2/2
File
A:\NEWCCT2.SCH
Drawn By:
Panupat Tansiri

```
unit ppp_project;

interface

uses
  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
  ExtCtrls, Grids, StdCtrls, Menus, Buttons, ComCtrls, ToolWin, ImgList;

type
  TForm1 = class(TForm)
    Button1: TButton;
    StringGrid1: TStringGrid;
    Timer1: TTimer;
    Edit1: TEdit;
    Button2: TButton;
    Button3: TButton;
    Edit2: TEdit;
    Button4: TButton;
    Timer2: TTimer;
    Timer3: TTimer;
    Edit3: TEdit;
    Edit4: TEdit;
    Label1: TLabel;
    Label2: TLabel;
    Timer4: TTimer;
    Timer5: TTimer;
    Label3: TLabel;
    Label4: TLabel;
    OpenDialog1: TOpenDialog;
    SaveDialog1: TSaveDialog;
    ImageList1: TImageList;
    ToolBar1: TToolBar;
    ToolButton1: TToolButton;
    ToolButton2: TToolButton;
    Button5: TButton;
    Edit5: TEdit;
    Label5: TLabel;
    Label6: TLabel;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    Button6: TButton;
    Edit6: TEdit;
    Edit7: TEdit;
    Label7: TLabel;
    Label8: TLabel;
    Label9: TLabel;
    Label10: TLabel;
    Label11: TLabel;
    Button7: TButton;
    Timer6: TTimer;
    Timer7: TTimer;
    Button8: TButton;
    Button9: TButton;
    Timer8: TTimer;
    Button10: TButton;
    Button11: TButton;
    Button12: TButton;
    Button13: TButton;
    Timer9: TTimer;
    Timer10: TTimer;
    Timer11: TTimer;
    Timer12: TTimer;
    Button14: TButton;
    procedure Button1Click(Sender: TObject);
    procedure Timer1Timer(Sender: TObject);
    procedure Button2Click(Sender: TObject);
    procedure Button4Click(Sender: TObject);
    procedure Timer2Timer(Sender: TObject);
    procedure Timer3Timer(Sender: TObject);
    procedure Timer4Timer(Sender: TObject);
    procedure Timer5Timer(Sender: TObject);
    procedure StringGrid1RowMoved(Sender: TObject; FromIndex,
      ToIndex: Integer);
    procedure ToolButton1Click(Sender: TObject);
    procedure ToolButton2Click(Sender: TObject);
    procedure Button5Click(Sender: TObject);
    procedure Button3Click(Sender: TObject);
    procedure Button6Click(Sender: TObject);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    procedure Button7Click(Sender: TObject);
    procedure Timer6Timer(Sender: TObject);
    procedure Timer7Timer(Sender: TObject);
    procedure Button8Click(Sender: TObject);
    procedure Button9Click(Sender: TObject);
    procedure Timer8Timer(Sender: TObject);
    procedure Button10Click(Sender: TObject);
    procedure Button12Click(Sender: TObject);
    procedure Timer12Timer(Sender: TObject);
    procedure Button13Click(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  Form1: TForm1;
  row,total1,total2,x,y,x1,y1,mv11,mv12,mv21,mv22,switch,step,error,j:integer;
  c1,c2: array[1..50] of integer;
  data: array[0..3,0..3] of integer;
  m,n,dx,dy,x0,y0:integer;
  rstep,rerror,x12:integer;
  punch,back,xs,ys:byte;
  xc,yc,xr,yr:real;

implementation
uses unit2;



{$R *.DFM}

procedure portout(ioport:word;value:byte);stdcall;
        asm
        mov dx,[ioport];
        mov al,value;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        out dx,al;
        end;
function portin(ioport:word):integer;stdcall;
       asm
        mov dx,[ioport];
        in al,dx;
       end;
procedure TForm1.Button1Click(Sender: TObject);//Check data right or wrong
begin
 for row:=0 to 50 do
    begin
    if stringgrid1.cells[0,row]='end' then
    begin
    total1:=total1+1;
    end;
    end;
    row:=0;
    if total1=1 then
    begin
    while stringgrid1.cells[0,row]<>'end' do
    begin
    xc:=strtofloat(stringgrid1.cells[0,row]);
    yc:=strtofloat(stringgrid1.cells[1,row]);
    if ((xc<0) or (xc>15)) or ((yc<0) or (yc>10)) then
    begin
    total2:=total2+1;
    end;
    row:=row+1;
    end;
    end;
    if total1=1 then
    begin
    if total2=0 then
    begin
    edit4.text:='True condition';
    timer1.enabled:=true;
    end
    else
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      begin
      MessageDlg('(x,y) is out of the range',mtError,[mbOk],0);
      timer1.enabled:=false;
      end;
      end
      else

      begin
      MessageDlg('Please write end at the last row',mterror,[mbOk],0);
      timer1.enabled:=false;
      end;
      xc:=0;
      yc:=0;
      row:=0;
      total1:=0;
      total2:=0;
end;

procedure TForm1.Timer1Timer(Sender: TObject);//Recieve value from stringgrid1
begin
   timer1.Enabled:=false;
   edit3.text:='row =  '+inttostr(j);
   data[0,0]:=17;
   data[0,1]:=18;
   data[0,2]:=20;
   data[0,3]:=24;
   data[1,0]:=33;
   data[1,1]:=34;
   data[1,2]:=36;
   data[1,3]:=40;
   data[2,0]:=65;
   data[2,1]:=66;
   data[2,2]:=68;
   data[2,3]:=72;
   data[3,0]:=129;
   data[3,1]:=130;
   data[3,2]:=132;
   data[3,3]:=136;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
if stringgrid1.cells[0,j]<>'end' then
 begin
 button2.enabled:=false;
 xr:=strtofloat(stringgrid1.cells[0,j]);
 yr:=strtofloat(stringgrid1.cells[1,j]);
 xr:=(1000/3.466)*xr;
 yr:=(1000/2.2)*yr;
 x1:=round(xr);
 y1:=round(yr);
 dx:=x1-x0;
 dy:=y1-y0;
 x:=dx;
 y:=dy;
 x0:=x1;
 y0:=y1;
 mv11:=1;
 mv12:=1;
 mv21:=1;
 mv22:=1;
 if x<0 then
 begin
 mv11:=-1;
 mv21:=-1;
 end;
 if y<0 then
 begin//3
 mv12:=-1;
 mv22:=-1;
 end;
 if abs(x)>=abs(y) then mv22:=0
 else
 begin
 mv21:=0;
 switch:=x;
 x:=y;
 y:=switch;
 end;
 step:=abs(x)+abs(y);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      error:=-abs(x);
      x:=2*abs(x);
      y:=-2*abs(y);
      timer2.enabled:=true;
      end
      else
      begin
      timer3.enabled:=false;
      timer2.enabled:=false;
      timer4.enabled:=false;
      timer5.enabled:=false;
      timer12.enabled:=false;
      end;
end;
procedure TForm1.Button2Click(Sender: TObject);//Stop motor
begin
     timer1.enabled:=false;
end;

procedure TForm1.Timer2Timer(Sender: TObject);//Motor Run
begin
    if step>0 then
    begin
    error:= error-y;
    if error>0 then
    begin
    m:=m+mv11;
    n:=n+mv12;
    if m=-1 then m:=3;
    if n=-1 then n:=3;
    if m=4  then m:=0;
    if n=4  then n:=0;
    edit1.text:=inttostr(m);
    edit2.text:=inttostr(n);
    edit4.text:=inttostr(step);
    portout(888,data[m,n]);
    error:=error-x;
    step:=step-2;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    end
    else
    begin
    m:=m+mv21;
    n:=n+mv22;
    if m=-1 then m:=3;
    if n=-1 then n:=3;
    if m=4  then m:=0;
    if n=4  then n:=0;
    edit1.text:=inttostr(m);
    edit2.text:=inttostr(n);
    edit4.text:=inttostr(step);
    portout(888,data[m,n]);
    step:=step-1;
    end;

    end
    else
    begin
        timer2.enabled:=false;
        edit4.text:='Piston out';
        portout(890,9);//piston out
        timer4.enabled:=true;
    end;
end;

procedure TForm1.Button4Click(Sender: TObject);//Set x,y = 0
begin
    portout(890,2);
    edit4.text:='x='+inttostr(x1)+'  '+'y='+inttostr(y1);
    mv11:=-1;
    mv12:=-1;
    mv21:=-1;
    mv22:=-1;
    if abs(x1)>=abs(y1) then mv22:=0
    else
    begin
    mv21:=0;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    switch:=x1;
    x1:=y1;
    y1:=switch;
    end;
    rstep :=abs(x1)+abs(y1);
    rerror:=-abs(x1);
    x1:=2*abs(x1);
    y1:=-2*abs(y1);
    timer3.Enabled:=true;
end;

procedure TForm1.Timer3Timer(Sender: TObject);//Set x,y = 0
begin
    edit3.text:=inttostr(rstep)+'  '+inttostr(rerror);
    if rstep>0 then
    begin
    rerror:=rerror-y1;
    if rerror>0 then
    begin
    m:=m+mv11;
    n:=n+mv12;
    if m=-1 then m:=3;
    if n=-1 then n:=3;
    if m=4  then m:=0;
    if n=4  then n:=0;
    edit1.text:=inttostr(m);
    edit2.text:=inttostr(n);
    portout(888,data[m,n]);
    rerror:=rerror-x1;
    rstep:=rstep-2;
    end
    else
    begin
    m:=m+mv21;
    n:=n+mv22;
    if m=-1 then m:=3;
    if n=-1 then n:=3;
    if m=4  then m:=0;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      if n=4  then n:=0;
      edit1.text:=inttostr(m);
      edit2.text:=inttostr(n);
      portout(888,data[m,n]);
      rstep:=rstep-1;
      end;

      end
      else
      begin
      edit4.text:='finish';
      timer3.enabled:=false;
      j:=0;
      timer2.enabled:=false;
      timer4.enabled:=false;
      timer5.enabled:=false;
      timer12.enabled:=false;
      edit3.text:='row =  '+inttostr(j);
      x1:=0;
      y1:=0;
      x0:=0;
      y0:=0;
      dx:=0;
      dy:=0;
      x:=0;
      y:=0;
      step:=0;
      error:=0;
      mv11:=1;
      mv12:=1;
      mv21:=1;
      mv22:=1;
    end;
end;

procedure TForm1.Timer4Timer(Sender: TObject);//Check piston position
begin
     punch:=portin(889);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        if punch=88 then
        begin
        timer4.enabled:=false;
        timer12.enabled:=true;
        punch:=0;//Reset punch
        end;
 end;

 procedure TForm1.Timer12Timer(Sender: TObject);//Return piston
 begin
     timer12.enabled:=false;
     edit4.text:='Piston  back';
     portout(890,2);//piston in
     timer5.enabled:=true;
 end;

 procedure TForm1.Timer5Timer(Sender: TObject);//Check piston position and Run motor
 begin
     back:=portin(889);
     if back<>88 then
     begin
     timer5.enabled:=false;
     edit4.text:='Piston back complete';
     j:=j+1;
     timer1.enabled:=true;
     button2.enabled:=true;
     end;
end;

procedure TForm1.StringGrid1RowMoved(Sender: TObject; FromIndex,
  ToIndex: Integer);
begin
    label3.caption:=inttostr(abs(FromIndex-ToIndex));
end;

procedure TForm1.ToolButton1Click(Sender: TObject);//Save data
var F:TEXTFILE;
   i,j,k:integer;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  begin
      if Savedialog1.execute then
      begin
      caption:=ExtractFilename(opendialog1.Filename);
      AssignFile(F,SaveDialog1.Filename);
      Rewrite(F);
        for i:=0 to stringgrid1.RowCount do
        begin
        write(F,stringgrid1.cells[0,i]);
        j:=length(stringgrid1.cells[0,i]);
        j:=20-j;
           for k:=1 to j do
           write(F,' ');

        write(F,stringgrid1.cells[1,i],'    ');
        j:=length(stringgrid1.cells[1,i]);
        j:=20-j;
          for k:=1 to j do
          write(F,' ');
          writeln(F);
        end;
        closefile(F);
     end;
end;

procedure TForm1.ToolButton2Click(Sender: TObject);//Open data
var F:textfile;
    i,j:integer;
    a:array[0..1,0..60] of string[20];
label end1,end2;
begin
     if opendialog1.execute then
     begin
     caption:=ExtractFilename(opendialog1.Filename);
     AssignFile(F,openDialog1.Filename);
     Reset(F);
       for i:=0 to stringgrid1.RowCount do
       begin
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        read(F,a[0,i]);
         for j:=1 to 20 do
         if a[0,i][j]=' ' then
         begin
         delete(a[0,i],j,20);
         goto end1;
         end;
        end1:

        read(F,a[1,i]);
         for j:=1 to 20 do
         if a[1,i][j]=' ' then
         begin
         delete(a[1,i],j,20);
         goto end2;
         end;
        end2:

        readln(F);

        end;

       closefile(F);
    end;

    for i:=0 to stringgrid1.rowcount do
    begin
    stringgrid1.cells[0,i]:=a[0,i];
    stringgrid1.cells[1,i]:=a[1,i];
    end;
end;

procedure TForm1.Button5Click(Sender: TObject);//Draw
var i,j,plot,rowplot:integer;
   x,y:array[0..50] of real;
   xc,yc:real;
begin
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Form2.close;
plot:=0;
rowplot:=0;
for row:=0 to 50 do
begin//1
if stringgrid1.cells[0,row]='end' then
begin//2
total1:=total1+1;
end;//2
end; //1
row:=0;
if total1=1 then
begin//3
while stringgrid1.cells[0,row]<>'end' do
begin//4
xc:=strtofloat(stringgrid1.cells[0,row]);
yc:=strtofloat(stringgrid1.cells[1,row]);
if ((xc<0) or (xc>1200)) or ((yc<0) or (yc>1200)) then
begin//5
total2:=total2+1;
end;//5
row:=row+1;
end;//4
end;//3
if total1=1 then
begin//6
if total2=0 then
begin
edit4.text:='True condition';
rowplot:=row-1;
plot:=1;
end
else
begin
MessageDlg('(x,y) is out of the range',mtError,[mbOk],0);
plot:=0;
end;
end//6
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
else
begin//7
MessageDlg('Please write end at the last row',mterror,[mbOk],0);
timer1.enabled:=false;
end;//7
xc:=0;
yc:=0;
row:=0;
total1:=0;
total2:=0;
if plot=1 then
begin
Form2.show;
Form2.canvas.pen.color:=clblack;
Form2.canvas.pen.width:=1;
Form2.canvas.moveto(20,350);
Form2.canvas.LineTo(20,10);
Form2.canvas.moveto(440,350);
Form2.canvas.LineTo(20,350);
Form2.canvas.moveto(440,350);
Form2.canvas.LineTo(440,10);
Form2.canvas.moveto(440,10);
Form2.canvas.LineTo(20,10);
end;
if plot=1 then
begin
for j:=0 to rowplot do
begin
x[j]:=strtofloat(stringgrid1.cells[0,j]);
x[j]:=30*x[j];
y[j]:=strtofloat(stringgrid1.cells[1,j]);
y[j]:=27*y[j];
end;
for j:=0 to rowplot do
begin
Form2.canvas.pen.color:=clblack;
Form2.canvas.pen.width:=6;
Form2.canvas.ellipse(20+round(x[j]),350-round(y[j]),20+round(x[j])+3,350-round(y[j])+3);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      end;
      end;
end;

procedure TForm1.Button3Click(Sender: TObject);//Step motor speed
var speed:integer;
begin
      speed:=strtoint(edit5.text);
      timer2.interval:=speed;
      timer3.interval:=speed;
end;

procedure TForm1.Button6Click(Sender: TObject);//Deflaut motor speed
begin
      edit5.text:='300';
      timer2.interval:=300;
      timer3.interval:=300;
end;

procedure TForm1.Button7Click(Sender: TObject);//Set limit switch
begin
      timer6.enabled:=true;
      x1:=0;
      y1:=0;
      x0:=0;
      y0:=0;
      dx:=0;
      dy:=0;
      x:=0;
      y:=0;
      step:=0;
      error:=0;
      mv11:=1;
      mv12:=1;
      mv21:=1;
      mv22:=1;
      data[0,0]:=17;
      data[0,1]:=18;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      data[0,2]:=20;
      data[0,3]:=24;
      data[1,0]:=33;
      data[1,1]:=34;
      data[1,2]:=36;
      data[1,3]:=40;
      data[2,0]:=65;
      data[2,1]:=66;
      data[2,2]:=68;
      data[2,3]:=72;
      data[3,0]:=129;
      data[3,1]:=130;
      data[3,2]:=132;
      data[3,3]:=136;
      timer3.enabled:=false;
      j:=0;
      timer2.enabled:=false;
      timer4.enabled:=false;
      timer5.enabled:=false;
      timer12.enabled:=false;
end;

procedure TForm1.Timer6Timer(Sender: TObject);//X set
begin
     xs:=portin(889);
     if xs<>112 then
     begin
     m:=m-1;
     if m=-1 then m:=3;
     if n=-1 then n:=3;
     portout(888,data[m,n]);
     end
     else
     begin
     timer6.enabled:=false;
     edit4.text:='Limit set x=0 Ok';
     timer7.enabled:=true;
     xs:=0;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      end;
  end;

  procedure TForm1.Timer7Timer(Sender: TObject);//Y set
  begin
      ys:=portin(889);
      if ys<>240 then
      begin
      n:=n-1;
      if n=-1 then n:=3;
      if m=-1 then m:=3;
      portout(888,data[m,n]);
      end
      else
      begin
      if m=3 then m:=-1;
      if n=3 then n:=-1;
      timer7.enabled:=false;
      portout(888,data[m+1,n+1]);
      ys:=0;
      edit4.text:='Limit switch set finsh';
      end;

 end;

 procedure TForm1.Button8Click(Sender: TObject);//Stop limit set
 begin
     timer6.enabled:=false;
     timer7.enabled:=false;
 end;

 procedure TForm1.Button9Click(Sender: TObject);//Clear stringgrid1 data
 begin
     timer8.enabled:=true;
     end;

 procedure TForm1.Timer8Timer(Sender: TObject);//Clear stringgrid1 data
 begin
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      if row<>stringgrid1.rowcount then
      begin
      stringgrid1.cells[0,row]:=' ';
      stringgrid1.cells[1,row]:=' ';
      row:=row+1;
      end
      else
      begin
      timer8.enabled:=false;
      row:=0;
      end;
 end;

 procedure TForm1.Button10Click(Sender: TObject); //Check addr 379 value
 begin
     x12:=10000;
     x12:=portin(889);
     edit4.text:=inttostr(x12);
 end;

 procedure TForm1.Button12Click(Sender: TObject);//Piston out
 var n1:integer;
 begin
     portout(890,9);
 end;

 procedure TForm1.Button13Click(Sender: TObject);//Piston in
 begin
     portout(890,2);
 end;

 end.
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้