บทที่ 2
หลักการและทฤษฎี

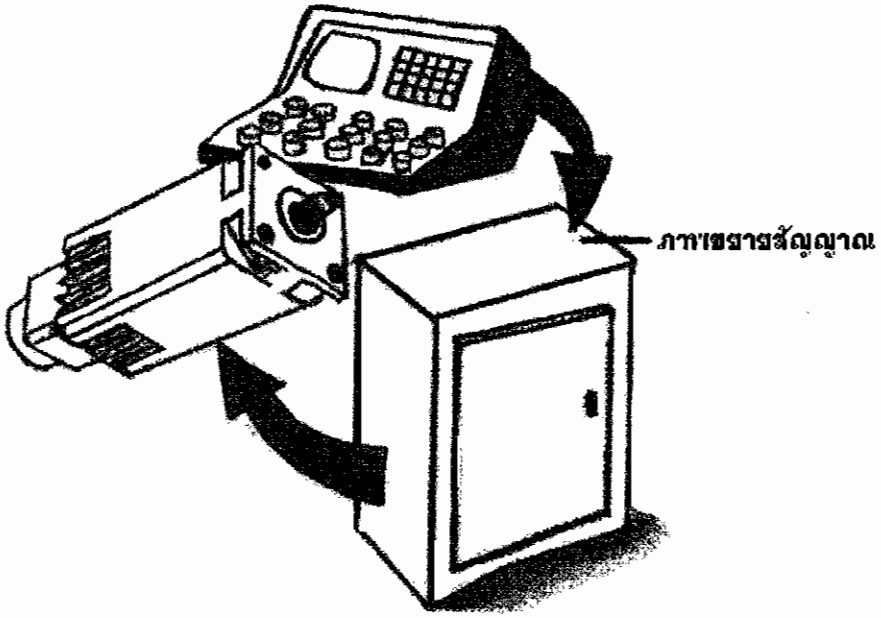
ทฤษฎีเกี่ยวข้องกับการทำโครงในเรื่องนี้ ส่วนมากจะเป็นทฤษฎีทางด้าน CNC เป็นส่วนมาก และเครื่องจักรกลที่ใช้ในการทำโครงงานนี้ก็คือ เครื่องกลึง ดังนั้นจึงขออ้างทฤษฎีเกี่ยวข้องดังนี้

2.1 การทำงานของเครื่องจักรกล NC

หลักการทำงานของเครื่องจักรกล NC หรือ CNC จะคล้ายคลึงกับเครื่องจักรกลทั่วไปกล่าวคือ โดยพื้นฐานเบื้องต้นแล้วเครื่องจักรกล NC จะทำงานผลิตชิ้นงานเหมือนกับเครื่องจักรกลทั่วๆไป เพียงแต่ระบบควบคุม NC ของเครื่องจะทำงานในขั้นตอนต่างๆ แทนช่างควบคุมเครื่อง แต่ก่อนที่เครื่องจักร NC จะสามารถทำงานได้นั้น จะต้องป้อนโปรแกรมเข้าไปในระบบควบคุมก่อน

เมื่อระบบควบคุมอ่านโปรแกรมที่ป้อนเข้าไปแล้ว ก็จะนำไปควบคุมเครื่องให้เครื่องจักรทำงาน แต่เครื่องจักร NC ไม่มีมือสำหรับหมุน ดังนั้นแท่นเลื่อนต่างๆ จะต้องมีมอเตอร์ป้อน (feed motor) ประกอบอยู่

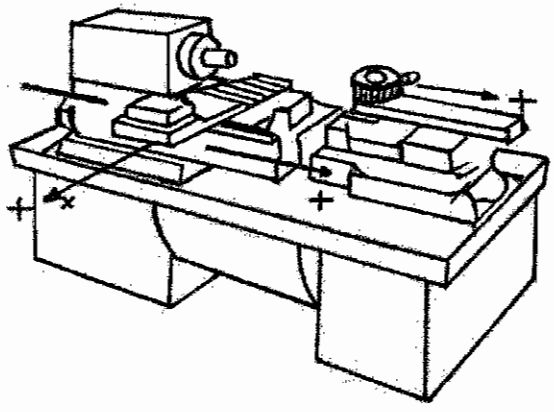
เมื่อระบบควบคุมอ่านโปรแกรมแล้ว ก็จะเปลี่ยนรหัสโปรแกรมนั้นให้เป็นสัญญาณทางไฟฟ้าเพื่อไปควบคุมให้มอเตอร์ทำงาน แต่เนื่องจากสัญญาณที่ออกมาจากระบบควบคุมมีน้อยดังนั้นจึงต้องส่งสัญญาณนี้เข้าไปในภาคขยายสัญญาณของระบบขับ (drive amplified) และส่งต่อไปยังมอเตอร์ป้อนของแนวแกนที่ต้องการเคลื่อนที่[1]



รูปที่ 2.1 การส่งสัญญาณควบคุมการทำงานของมอเตอร์

2.2 เครื่องกลึง NC (NC Turning Machines)

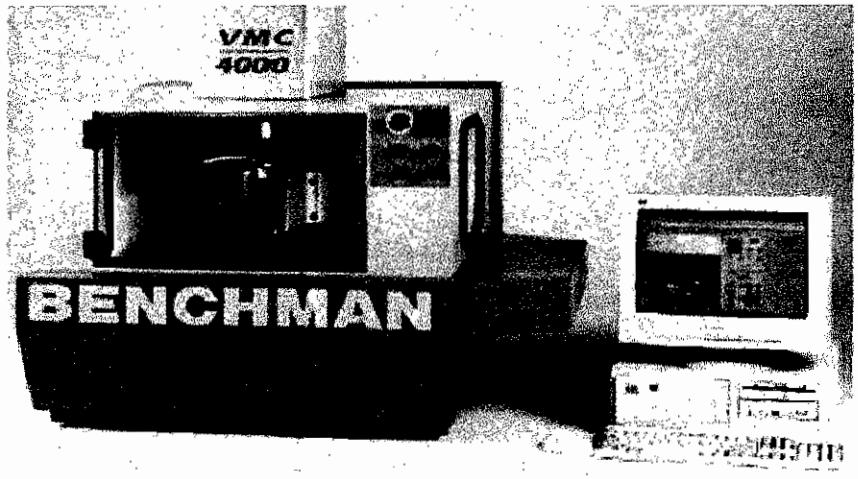
เครื่องกลึง NC ส่วนใหญ่จะมีแนวแกนการเคลื่อนที่ 2 – 3 แกน ลักษณะการออกแบบส่วนใหญ่จะขึ้นอยู่กับชนิดของการเครื่องกลึง[4] ดังรูป 2



รูปที่ 2.2 เครื่องกลึงยันศูนย์ NC

2.3 องค์ประกอบของเครื่องจักรที่ควบคุมได้

องค์ประกอบหรือชิ้นส่วนของเครื่องจักรกล ที่ทำหน้าที่เคลื่อนที่เข้าตัดเฉือนชิ้นงาน และองค์ประกอบอื่นๆที่ช่วยเสริมการทำงานตัดเฉือนให้สมบูรณ์ยิ่งขึ้น จะถูกควบคุมโดยโปรแกรม NC ด้วยวิธีการควบคุมแบบต่างๆ กัน[1]



รูปที่ 2.3 เครื่องกลึง CNC

ช่างชำนาญงานที่ทำหน้าที่ควบคุมการทำงานของเครื่องจักรกล NC หรือ CNC จะต้องคุ้นเคยกับหน้าที่การทำงานและขีดจำกัดในการทำงานของเครื่องจักรกล CNC นั้นเป็นอย่างดี ดังนั้นสิ่งที่จำเป็นที่ช่างต้องรู้ว่าองค์ประกอบส่วนใดของเครื่องจักรกล CNC สามารถควบคุมได้และมีวิธีการควบคุมอย่างไร องค์ประกอบของเครื่องจักรกล NC และ CNC ที่สามารถควบคุมได้แก่

- แนวแกนป้อน (Feed axes)
- การขับป้อน (Feed drives)
- อุปกรณ์วัดขนาด (Measuring devices)
- อุปกรณ์เปลี่ยนเครื่องมือตัด (Tool changers)
- แนวแกนหมุนและแนวแกนป้อนอื่นๆ

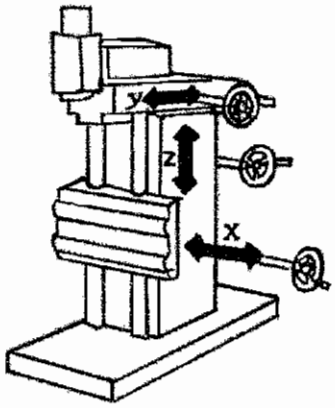
จะขอกล่าวถึงองค์ประกอบข้างต้นที่เกี่ยวข้องโดยตรงกับการทำโครงงานครั้งนี้เพียงเท่านั้น ซึ่งได้แก่[1]

2.3.1 แนวแกนป้อน (Feed axes)

ในการกล่าวถึงเครื่องจักรกล CNC บ่อยครั้งที่เราจะได้ยินคำว่า แนวแกน (axes) ซึ่งหมายถึงแนวการเคลื่อนที่ขององค์ประกอบของเครื่องจักรกล

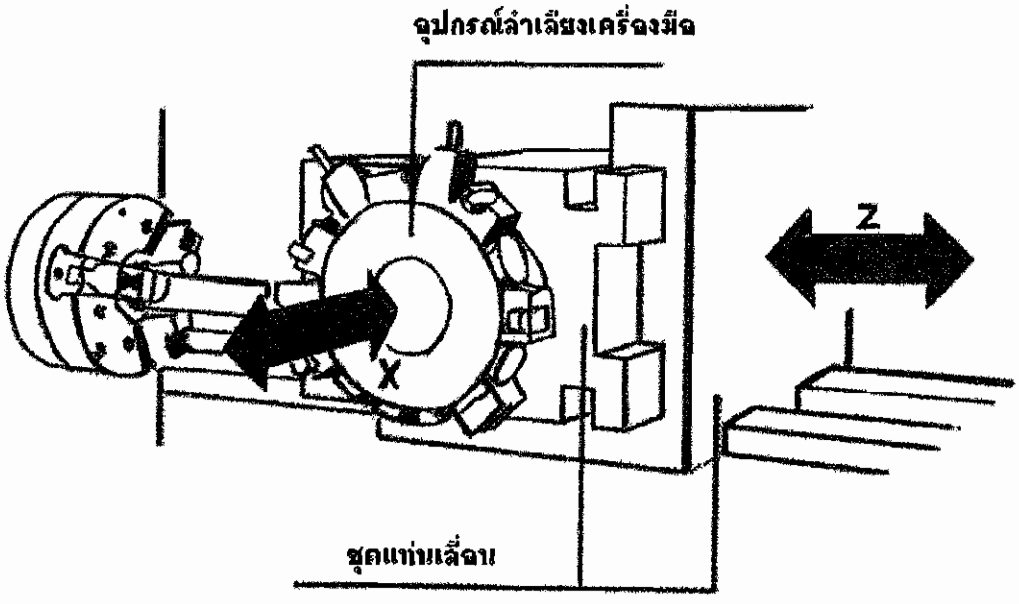
สำหรับเครื่องจักรกลทั่วๆไป การเคลื่อนที่ในแนวแกนต่างๆ จะเกิดจากการหมุนมือหมุนหรือโยกคันโยกป้อนอัตโนมัติ (Feed levers)

เครื่องจักกล CNC จะมีแนวแกนป้อนรวมกันอยู่หลายแนวแกน ทำให้สามารถตัดชิ้นงานให้เป็นรูปทรงต่างๆ ที่ต้องการได้ การกำหนดแนวแกนต่างๆ ของเครื่องจักรกล CNC จะกำหนดตามมาตรฐานสากล ซึ่งจะกำหนดแนวแกนเหล่านี้โดยใช้ตัวอักษร x, y และ z ดังรูป



รูปที่ 2.4 แท่งเลื่อนแบบ 3 แกน

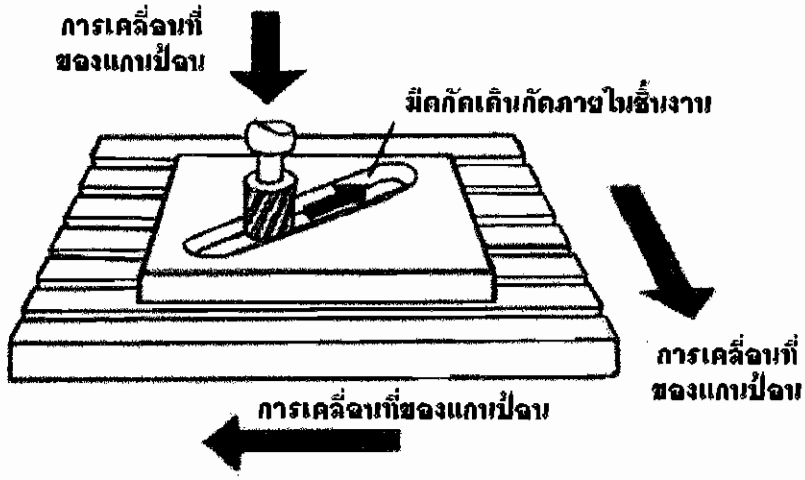
เครื่องกลึงจะมีแนวแกนป้อนอยู่ 2 แนวแกน คือ xและ z ทั้งสองแกนจะอยู่ที่แท่นเลื่อนซึ่งมีอุปกรณ์ลำเลียงเครื่องมือ ติดตั้งอยู่ลักษณะเช่นนี้ทำให้สามารถกลึงงานที่มีรูปทรงต่าง ๆกันได้ตามต้องการ



รูปที่ 2.5 เครื่องกลึง CNC แบบ 2 แกน

2.3.2 การขับป้อน (Feed drives)

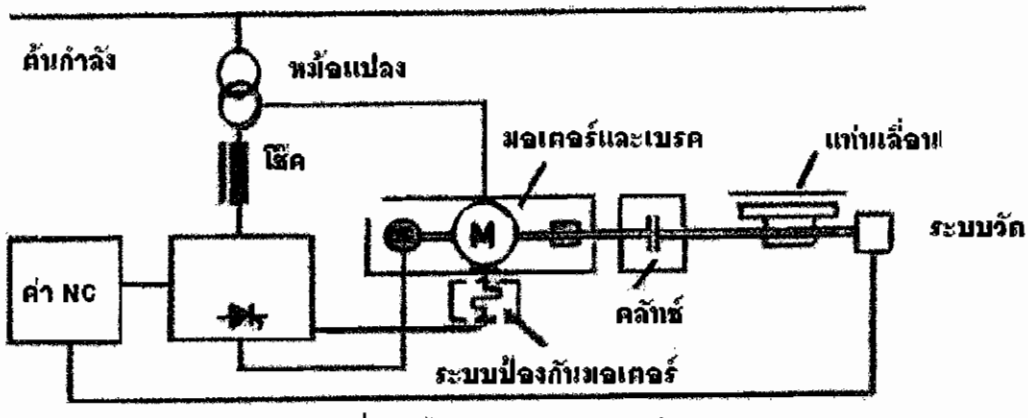
การเคลื่อนที่เรียงลำดับกันหรือพร้อมๆ กันอย่างต่อเนื่องของแนวแกนป้อน จะทำให้เกิดการตัดเฉือนของเครื่องมือในชิ้นงานดังรูป 2.6



รูปที่ 2.6 การเคลื่อนที่ตัดเฉือนของเครื่องมือตัด

การขับป้อนจะทำให้เกิดการเคลื่อนที่ของแท่นเลื่อน ในขณะตัดเฉือนแท่นเลื่อนอาจพาให้ชิ้นงานเคลื่อนที่หรือคมตัดเคลื่อนที่ก็ได้

ระบบขับป้อนโดยทั่วไปจะใช้มอเตอร์กระแสตรงในการขับหมุนและควบคุมการทำงานด้วยวงจรอิเล็กทรอนิกส์จากภายนอก มอเตอร์ชนิดนี้จะสามารถหมุนและเบรกได้ทั้งสองทิศทาง ขณะตัดเฉือนชิ้นงาน การเคลื่อนที่ป้อนจะต้องเป็นไปอย่างสม่ำเสมอ และสามารถต้านแรงกระทำจากภายนอกได้ ด้วยเหตุนี้ระบบขับป้อนจึงต้องได้รับการออกแบบให้มีความแข็งแกร่งสูง มีการเคลื่อนที่คงที่และสม่ำเสมอ สามารถตอบสนองต่อการเปลี่ยนอัตราป้อนได้อย่างรวดเร็ว ปัจจัยหนึ่งที่จะทำให้ระบบขับป้อนสามารถทำงานได้อย่างมีประสิทธิภาพก็คือ การเลือกใช้อุปกรณ์ในระบบขับป้อนให้เหมาะสมกับการทำงานของเครื่องจักร และการออกแบบวงจรควบคุมการทำงานอย่างมีประสิทธิภาพ[1] ดังรูป 2.7



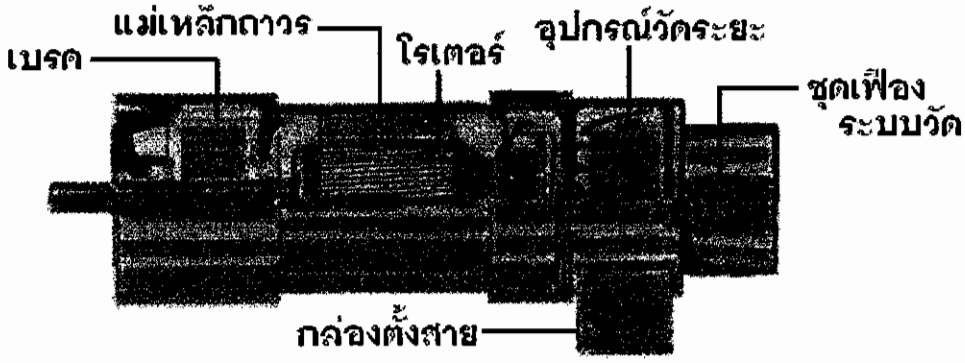
รูปที่ 2.7 ไดอะแกรมระบบขับป้อน

2.4 มอเตอร์

เครื่องจักกล NC สมัยใหม่จะออกแบบใช้ระบบขับป้อนแบบเซอโว (servo drives) ทำให้สามารถปรับอัตราป้อนและความเร็วรอบได้โดยไม่มีขีดจำกัดของขั้นความเร็วและอัตราป้อนมอเตอร์ที่ใช้ในระบบขับป้อนโดยทั่วไปจะมีอยู่ 3 ชนิด ด้วยกันคือ[1]

ก. มอเตอร์กระแสตรง (DC motors)

ลักษณะสร้างของมอเตอร์กระแสตรงจะใช้เป็นแม่เหล็กถาวรที่มี 4, 6 หรือ 8 ขั้วประกอบด้วยระบบเบรค (Brake) แกนมอเตอร์ (Rotor) อุปกรณ์วัดรอบ (Tachogenerator) และขั้วอุปกรณ์วัด (Measuring box) ดังรูป 2.8



รูปที่ 2.8 ส่วนประกอบของมอเตอร์กระแสตรง

การใช้มอเตอร์กระแสตรง ทำให้สามารถปรับอัตราป้อนได้ละเอียดและมีวงจรควบคุมที่ไม่ซับซ้อน แต่มีข้อเสียคือ ตรงที่มอเตอร์ชนิดนี้ต้องใช้แปรงถ่าน ซึ่งจะต้องทำความสะอาดและเปลี่ยนเมื่อแปรงถ่านหมด นอกจากนี้แปรงถ่ายยังทำให้แกนมอเตอร์สึกหรอ อันเป็นผลทำให้กำลังมอเตอร์ลดลง ข้อเสียอีกประการหนึ่งก็คือ หากต้องการกำลังขับสูง มอเตอร์ก็จะมีขนาดใหญ่ด้วยและเมื่อใช้ความเร็วรอบสูงๆ จะทำให้แรงบิดลดลง ดังนั้น จึงมักใช้กับเครื่องจักรกล NC ขนาดเล็กและขนาดกลาง

ข. มอเตอร์แบบเป็นขั้น (stepping motors)

เป็นมอเตอร์ที่ทำงานแบบต่อเนื่องโดยการแปลงคลื่นสัญญาณที่ป้อนเข้าไปในระบบให้เป็นการเคลื่อนที่เชิงมุม การหมุนในแต่ละมุมหรือขั้นที่เปลี่ยนไป 1 ขั้นจะเท่ากับ 1 คลื่นสัญญาณ ดังนั้น ตำแหน่งของเพลาจะถูกกำหนดโดยคลื่นสัญญาณที่ป้อนเข้าไปในระบบและความสำเร็จในการหมุนของเพลาจะวัดเป็นจำนวนขั้นต่อวินาที (steps/second) ซึ่งเท่ากับความถี่ของคลื่นสัญญาณที่ป้อนเข้าในระบบที่วัดเป็นจำนวนคลื่นสัญญาณต่อวินาที (pulses/second) ความเที่ยงตรงของระบบจะขึ้นอยู่กับความสามารถของมอเตอร์ชนิดนี้จะลดลงเมื่อความเร็วในการหมุนแบ่งเพิ่มขึ้น ดังนั้น จึงเหมาะสำหรับเครื่องจักรกลเล็กๆ ที่ไม่ต้องใช้กำลังขับมาก

ค. มอเตอร์กระแสสลับ (Alternate-current motors)

ส่วนมากจะเป็นมอเตอร์แบบ Synchronous motor ข้อดีของมอเตอร์ชนิดนี้คือ ไม่ต้องใช้แปรงถ่าน ทำให้สามารถลดงานบำรุงรักษาได้มากและมอเตอร์ขนาดเดียวกันเมื่อเปรียบเทียบกับมอเตอร์แบบนี้คือ วงจรควบคุมจะมีความซับซ้อนมากกว่าวงจรควบคุมมอเตอร์กระแสตรง

2.5 การจัดการในระบบ NC

2.5.1 วิธีการจัดเตรียมโปรแกรม

การจัดเตรียมโปรแกรม NC มีวิธีการที่แตกต่างกันหลายวิธีขึ้นอยู่กับการจัดการเกี่ยวกับวิธีการทำโปรแกรมของแต่ละบริษัทและอุปกรณ์ที่ใช้ช่วยในการประมวลผลข้อมูล[4]

การจัดเตรียมโปรแกรมในฝ่ายของแผนและโรงงาน

การจัดเตรียมโปรแกรมนั้นวิธีการที่แตกต่างกันอยู่ 2 วิธี คือการจัดเตรียมโปรแกรมจากฝ่ายวางแผนกับการจัดเตรียมโปรแกรมในโรงงาน ซึ่งขึ้นอยู่ว่าการทำโปรแกรมนั้นกระที่ไหน ทั้งสองวิธีนี้ต่างก็มีข้อพิจารณาในการเลือกใช้ต่างๆ กัน

การพิจารณาเลือกใช้วิธีการทำโปรแกรมจากฝ่ายวางแผนมีข้อพิจารณาดังนี้ คือ

- ระบบ CNC ที่ใช้อยู่มีความซับซ้อนมาก
- ชิ้นงานที่มีรูปทรงเลขาคณิตมีความซับซ้อนและยุ่งยากในการทำโปรแกรม
- มีเครื่องจักรกล CNC ที่คล้ายคลึงกันเป็นจำนวนมากในโรงงาน
- บุคลากรในโรงงานมีความสามารถไม่เพียงพอ
- สามารถหาระบบช่วยในการทำโปรแกรมได้

ส่วนการเลือกพิจารณาเลือกใช้วิธีการทำโปรแกรมในโรงงานจะมีข้อพิจารณาเลือกใช้ดังนี้

- เครื่องจักรกล CNC ที่ใช้สามารถควบคุมการทำงานได้ง่าย
- ชิ้นงานมีรูปทรงเรขาคณิตคล้ายคลึงกัน
- ในโรงงานมีเครื่องจักรกล CNC จำนวนน้อย
- บุคลากรในโรงงานมีความสามารถสูงเพียงพอ
- การแก้ไขข้อผิดพลาดและการปรับปรุงโปรแกรมสามารถทำได้รวดเร็ว

รูปที่ 2.9 การโปรแกรมในโรงงานโดยมีการช่วยนำ

เครื่องจักรกล CNC ส่วนมากจะออกแบบไว้สำหรับการทำโปรแกรมในโรงงานซึ่งก็มีความแตกต่างในการควบคุมมากเช่นกัน เพื่อช่วยช่างควบคุมเครื่องในโรงงานทำโปรแกรม NC ดังนั้น ระบบ CNC ที่มีการแนะนำช่วงควบคุมนี้ บนจอภาพจะแสดงหน้าที่การทำงานที่สามารถเลือกใช้ได้เป็นขั้นๆ หรืออย่างต่อเนื่องในทุกสภาวะ และเมื่อป้อนโปรแกรมจอภาพไม่เพียงแต่แสดงคำสั่งที่ป้อนเข้าไปเท่านั้น แต่ยังแสดงเงื่อนไขเสริม เช่น ค่าโคออดเนตที่ต้องใช้ อัตราการป้อนเป็นต้น ดังแสดงในรูป 2.9

นอกจากนี้ยังมีสิ่งที่จะช่วยควบคุมเครื่องเกี่ยวกับการทำโปรแกรม NC คือ หมวดการทำงานที่เรียกว่า การปฏิบัติย้อนหลัง (play back) และการสอนงาน (teach-in)

ในการตัดเฉือนชิ้นงานบางลักษณะอาจเกี่ยวข้องกับขั้นตอนบางขั้นตอนที่ไม่สะดวกในการโปรแกรมด้วยค่าโคออดิเนต สำหรับในกรณีเช่นนี้จะมีวิธีการปฏิบัติย้อนหลังช่วยในการทำโปรแกรม

นอกจากนี้เครื่องจักรกล CNC ส่วนมากยังมีหมวดการทำงานที่เรียกว่าการสอนงาน (teach-in) ซึ่งสามารถใช้ได้เมื่อต้องการปรับตั้งจุดศูนย์และการจำกัดพื้นที่ทำงาน โดยการเลื่อนเครื่องมือให้เคลื่อนที่ไปยังตำแหน่งที่ต้องการ และป้อนเก็บบันทึกข้อมูลนั้นโดยตรง

วิธีสอนงานนี้มีการใช้บ่อยครั้งมากเพื่อกำหนดพื้นที่ทำงานที่สัมพันธ์กับอุปกรณ์จับยึดชิ้นงาน เช่นบริเวณพื้นที่ที่เครื่องมือจะต้องไม่เคลื่อนที่เข้าไป เพราะอาจเกิดการปะทะหรือชนกับอุปกรณ์จับยึดชิ้นงาน[4]

2.5.2 การโปรแกรมด้วยมือและการโปรแกรมด้วยคอมพิวเตอร์

การจัดทำโปรแกรม NC มีวิธีการทำที่สามารถทำได้อยู่ 2 วิธี คือ

- การจัดทำด้วยมือ
- การจัดทำด้วยคอมพิวเตอร์

EXAPT	TC-APT	ELAN	RWT/TRAUB/T2000	PHILIP II
MITURN	COMPACT II	MINI APT	PROGRAMAT	DIA-PROS

รูปที่ 2.10 ภาษาโปรแกรม NC ระดับสูง

2.6 ภาษาโปรแกรมเอ็นซี

ภาษาโปรแกรมของระบบควบคุม จะเป็นกฎที่ใช้ในการกำหนดว่าโปรแกรมล็อกใดบ้างที่จะต้องเขียนขึ้นสำหรับโปรแกรมเอ็นซี รายละเอียดของภาษาโปรแกรมที่ใช้ระบบควบคุมซีเอ็นซีจะมีการกำหนดเป็นมาตรฐาน โปรแกรมบล็อกจะประกอบด้วยจำนวนจำ (Words) หลายคำรวมกัน คำเหล่านี้จะประกอบขึ้นจากตัวอักษรหรือสัญลักษณ์ตัวเลขรวมกัน

คำที่ใช้ในโปรแกรมบล็อกอาจจะทำหน้าที่เป็นคำสั่ง หรือเป็นเงื่อนไขเสริมสำหรับการทำงานก็ได้ขึ้นอยู่กับตัวอักษรและตัวเลขที่กำกับอยู่ ตัวอักษรคำสั่งที่มีความสำคัญมาก คือ G คำสั่ง G (G00-G99) ส่วนมากจะเป็นคำสั่งที่ใช้เกี่ยวกับการควบคุมการเคลื่อนที่ของเครื่องมือ

ส่วนอักษรที่ใช้สำหรับเงื่อนไขเสริมที่สำคัญได้แก่

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

เมื่อโปรแกรมเอ็นซีถูกป้อนเข้าไปในระบบควบคุมแล้ว ระบบความคุมจะทำการตรวจสอบว่าการเขียนโปรแกรมนั้นเป็นไปตามกฎทั่วไปที่ใช้หรือไม่ ส่วนการป้อนข้อมูลโคออดิเนทที่ไม่ถูกต้องโดยช่างเขียนโปรแกรมการตรวจสอบจะพบได้เมื่อโปรแกรมทำงานแล้ว[2]

ตามมาตรฐานของเยอรมัน(DIN 66025) ความสำคัญของการใช้ตัวอักษร A – Z มีดังนี้

ตัวอักษร	ลักษณะสำคัญ
A	การหมุนรอบแกน x
B	การหมุนรอบแกน y
C	การหมุนรอบแกน z
D	หมายเลขการชดเชยขนาดเครื่องมือ
E	อัตราป้อนรอง
F	อัตราป้อน
G	คำสั่งการเคลื่อนที่
H	(ไม่มีการกำหนด)
I	พารามิเตอร์ของการแทนที่ หรือ ระยะพิตของเกลียวที่ขนานกับแกน X
J	พารามิเตอร์ของการแทนที่ หรือ ระยะพิตของเกลียวที่ขนานกับแกน Y
K	พารามิเตอร์ของการแทนที่ หรือ ระยะพิตของเกลียวที่ขนานกับแกน Z
L	(ไม่มีการกำหนด)
M	การทำงานเสริม
N	หมายเลขบล็อก
O	(ไม่มีการกำหนด)
P	การเคลื่อนที่ขนานกับแกน x แนวที่ 3
R	เคลื่อนที่เร็วในแกน Z หรือการเคลื่อนที่ขนานกับแกน Z แนวที่ 3
S	ความเร็วรอบของเพลางาน
T	เครื่องมือ
U	การเคลื่อนที่ขนานกับแกน X แนวที่ 2
V	การเคลื่อนที่ขนานกับแกน Y แนวที่ 2
W	การเคลื่อนที่ขนานกับแกน Z แนวที่ 2
X	การเคลื่อนที่ในแนวแกน X
Y	การเคลื่อนที่ในแนวแกน Y
Z	การเคลื่อนที่ในแนวแกน Z

ภาษาโปรแกรมเอ็นซีเป็นมาตรฐานสากล DIN 66025 การสร้างโปรแกรมสำหรับเครื่องจักรกลที่ควบคุมด้วยตัวเลข ซึ่งมีรายละเอียดตรงกันกับมาตรฐานสากล คือ ISO/DIS 6983 ISO /DP 6983 เครื่องจักรที่ควบคุมตัวเลข[2]

2.7 คำสั่งสำคัญในโปรแกรมเอ็นซี

ในหัวข้อนี้จะกล่าวถึงคำสั่งสำคัญๆ ที่ใช้กันเป็นมาตรฐานในภาษาโปรแกรมสำหรับระบบควบคุมเอ็นซี คำสั่งสำคัญเหล่านี้ได้แก่

G00 : การเคลื่อนที่เร็ว

G01 : การเคลื่อนที่แนวเส้นตรงตามค่าอัตราป้อน

G02 : การเคลื่อนที่แนวเส้นโค้งตามเข็มนาฬิกา

G03 : การเคลื่อนที่แนวเส้นโค้งทวนเข็มนาฬิกา

ถึงแม้ว่าผู้ผลิตระบบควบคุมเอ็นซีจะไม่ยึดมาตรฐานสากลในการใช้คำสั่งต่างๆ ก็ตาม กล่าวคือ ไม่ใช้คำสั่ง G00, G01, G02 และ G03 ก็จะต้องมีคำสั่งอื่นๆ ที่ทำให้เกิดผลเช่นเดียวกับคำสั่งดังกล่าว ซึ่งคำสั่งเหล่านี้อาจใช้อักษรที่แตกต่างออกไป หรือใช้เป็นสัญลักษณ์ในแป้นพิมพ์แทนก็ได้

ข้อมูลโคออดิเนทที่ใช้เป็นเงื่อนไขเสริม นอกเหนือจากคำสั่งที่กล่าวมาข้างต้นแล้วยังสามารถป้อนข้อมูลด้วยวิธีการต่างๆ กัน ขึ้นอยู่กับชนิดของระบบควบคุม

- ป้อนข้อมูลเป็นแบบการให้ขนาดสมบูรณ์
- ป้อนข้อมูลเป็นแบบการให้ขนาดต่อเนื่อง
- ป้อนข้อมูลเป็นแบมุมเสริม
- ป้อนข้อมูลบางส่วนเป็นโพลาร์โคออดิเนท

ด้วยเหตุผลที่ต้องการให้ระบบควบคุมเอ็นซีทำงานได้ง่ายจึงมีคำสั่งในโปรแกรมจำนวนหลายคนที่สามารถใช้ได้อย่างต่อเนื่อง จนกว่าจะมีการเปลี่ยนแปลงคำสั่งนั้นใหม่ กล่าวคือ คำสั่งเหล่านี้จะเป็นคำสั่งที่คงอยู่ในโปรแกรม และมีผลทำงานได้ตลอดไปหลังจากที่ป้อนโปรแกรมเข้าไปแล้ว จนกว่าจะมีการเปลี่ยนแปลงคำสั่งใหม่หรือเปลี่ยนแปลงเงื่อนไขเสริมใหม่[2]

2.7.1 คำสั่งการเคลื่อนที่เร็ว (Rapid traverse) : G00

คำสั่งที่ใช้ในการทำงานแบบเคลื่อนที่เร็ว จะกำหนดโปรแกรมด้วยคำสั่ง G00 คำสั่งนี้จะใช้ในการเคลื่อนที่ของเครื่องมือ เช่น มีดกัด มีดกลึง เป็นต้น ไปยังจุดเป้าหมายด้วยอัตราการเคลื่อนที่เร็วของเครื่องมือใช้คำสั่งนี้จะต้องมีเงื่อนไขเสริม คือจุดโคออดิเนทของจุดเป้าหมายที่ต้องการเคลื่อนเครื่องมือไป

โดยทั่วไปเส้นทางเดินของเครื่องมือ จะต่อเป็นเส้นตรงระหว่างจุดเริ่มต้นที่เรียกใช้คำสั่ง G00 กับจุดเป้าหมายที่กำหนดค่าโคออดิเนทไว้แล้ว

การเคลื่อนที่เร็วเป็นคำสั่งที่ใช้เฉพาะในการเคลื่อนที่ไปยังตำแหน่งที่ต้องการ คือ เครื่องมือต่างๆ จะต้องไม่สัมผัสกับชิ้นงาน[2]

2.7.2 การเคลื่อนที่แนวเส้นตรงตามอัตราป้อน (Straight-line at feedrate) : G01

การเคลื่อนที่ คือเส้นทางเดินของจุดปลายเครื่องมือในงานกลึง และเส้นทางเดินของจุดศูนย์กลางมีดกัดในงานกัด

ตามมาตรฐานที่กำหนดไว้ คำสั่งที่ใช้สำหรับการเคลื่อนที่แนวเส้นตรงตามค่าอัตราป้อนที่ใช้จะต้องใช้คำสั่งโปรแกรมคือ G01 ซึ่งจำเป็นต้องมีเงื่อนไขเสริมการทำงานดังนี้ : -

- ค่าโคออดิเนทของจุดเป้าหมาย
- อัตราป้อน
- ความเร็วรอบของเพลางาน หรือความเร็วตัด

เมื่อใช้คำสั่ง G01 เครื่องมือจะเคลื่อนที่เป็นแนวเส้นตรง ไปยังจุดเป้าหมายด้วยความเร็วตามค่าอัตราป้อนที่เลือกใช้

อัตราป้อนจะเป็นตัวกำหนดความเร็วที่ชิ้นงานถูกตัดเฉือนออก การเลือกใช้อัตราป้อนจะขึ้นอยู่กับเครื่องมือ (รูปทรงเรขาคณิตของจุดปลายเครื่องมือ วัสดุเครื่องมือ) วัสดุชิ้นงานที่ทำการเฉือน ผิวสำเร็จของชิ้นงาน ที่ต้องการ อัตราการขับเคลื่อน และความแข็งแกร่งของเครื่องจักรกล

การใช้คำสั่งการเคลื่อนที่แนวเส้นตรงตามค่าอัตราป้อนนี้ ในระบบควบคุมส่วนมากจะสามารถป้อนข้อมูลของตำแหน่งจุดเป้าหมายได้หลายแบบ เช่นเดียวกับการให้ขนาดแบบชิ้นงานจะสามารถใช้การให้ขนาดแบบสัมบูรณ์หรือแบบต่อเนื่องก็ได้ ด้วยเหตุผลนี้การป้อนตำแหน่งจุดเป้าหมายในระบบซีเอ็นซีจึงสามารถเลือกใช้เป็นแบบสัมบูรณ์หรือแบบต่อเนื่องได้ แต่ก่อนอื่นระบบควบคุมจะต้องรู้ว่าข้อมูลที่ป้อนนั้นเป็นแบบสัมบูรณ์หรือแบบต่อเนื่องเสียก่อน

ถ้าป้อนคำสั่งG90 ในโปรแกรม ค่าโคออดิเนทของจุดเป้าหมายในคำสั่งการเคลื่อนที่ทั้งหมดที่ป้อนข้อมูลต่อมา จะถูกระบบควบคุมอ่านค่าเป็นการให้ขนาดแบบสัมบูรณ์ และทันทีที่ป้อนคำสั่ง G91 เข้าไปในโปรแกรมระบบควบคุมก็จะเปลี่ยนการให้ขนาดเป็นแบบต่อเนื่อง

ระบบควบคุมส่วนมากจะอ่านค่าโคออดิเนท X, Y, Z เป็นค่าโคออดิเนทแบบสัมบูรณ์โดยอัตโนมัติ คือมีคำสั่ง G90 คงอยู่ในระบบความจำของระบบควบคุม ซึ่งในระบบนี้ค่าของ U, V, W จะใช้เป็นค่าโคออดิเนทแบบต่อเนื่อง[2]

2.7.3 การเคลื่อนที่แนวส่วนโค้งตามเข็มนาฬิกาและทวนเข็มนาฬิกา : G02/G03

คำสั่งการเคลื่อนที่แนวส่วนโค้งที่กำหนดเป็นมาตลบาน จะมีลักษณะการเคลื่อนที่ต่างกันระหว่างคำสั่ง G02 กับ G03 ซึ่งอยู่กับทิศทางการหมุน

การใช้คำสั่งการเคลื่อนที่แนวส่วนโค้งตามเข็มนาฬิกา (G02) และคำสั่งการเคลื่อนที่แนวส่วนโค้งทวนเข็มนาฬิกา (G03) จำเป็นต้องใช้ข้อมูลหรือเงื่อนไขเสริมดังนี้ : -

- ค่าโคออดิเนทของจุดเป้าหมาย
- ข้อมูลของขนาดรัศมีหรือจุดศูนย์กลางของส่วนโค้ง
- อัตราป้อน
- ความเร็วรอบของเพลางาน หรือความเร็วตัด

โดยทั่วไปจุดศูนย์กางของส่วนโค้ง มักจะป้อนข้อมูลเป็นการให้ขนาดแบบต่อเนื่องที่สัมพันธ์กับจุดเริ่มต้นโดยใช้อักษร I, J, และ K สำหรับระยะห่างในทิศทางของแนวแกน X, Y และ Z

I = ระยะห่างจากจุดเริ่มต้นของส่วนโค้งถึงจุดศูนย์กลางของส่วนโค้งในแนวแกน X

J = ระยะห่างจากจุดเริ่มต้นของส่วนโค้งถึงจุดศูนย์กลางของส่วนโค้งในแนวแกน Y

K = ระยะห่างจากจุดเริ่มของส่วนโค้งถึงจุดศูนย์กลางของส่วนโค้งในแนวแกน Z

การพิจารณาว่าเครื่องมือนั้นจะเคลื่อนที่ในทิศทางตามเข็มนาฬิกาหรือทวนเข็มนาฬิกาขึ้นอยู่กับทิศทางของแนวแกนที่ 3 ของระนาบการตัดเฉือนของรูปนั้น โดยใช้หลักของกฎมือขวา ซึ่งมองจากทิศทางของแนวแกนที่ 3 เป็นลบเสมอ

ตำแหน่งจุดศูนย์กลางของส่วนโค้งอาจอยู่ด้านซ้าย หรือด้านขวาของจุดเริ่มต้นก็ได้ดังนั้น การบอกค่าของ I, J และ K จึงต้องมีเครื่องหมายกำกับด้วย การพิจารณาเครื่องหมายนั้นมีหลักการง่ายๆ คือ เริ่มมองจากจุดเริ่มต้นไปหาจุดศูนย์กลางของส่วนโค้งในทิศทางที่ขนานกับแนวแกนที่ต้องการหาค่า ถ้าทิศทางที่มองไปนั้นตรงกับทิศทางของแนวแกนที่ต้องการหาค่าที่เป็นบวก ค่าของ I, J และ K เป็นรบวกด้วย แต่ถ้าตรงกับทิศทางของแนวแกนที่ต้องการหาค่าที่เป็นลบ ค่าของ I , J และ K เป็นลบด้วย[2]

2.8 คำสั่งการเขียนโปรแกรมการชดเชยขนาดตามเส้นขอบรูป

ในการเขียนโปรแกรมงานกัดที่ต้องการชดเชยขนาดตามเส้นขอบรูป จะมีคำสั่งให้เลือกใช้ตามลักษณะการเคลื่อนที่เดินกัดอยู่ 2 คำสั่ง คือ คำสั่ง G41 กับคำสั่ง G42 สำหรับการเรียกใช้โปรแกรมการชดเชยขนาดตามเส้นขอบรูป และใช้คำสั่ง G40 เพื่อยกเลิกการชดเชยขนาดตามเส้นขอบรูป

กฎในการพิจารณาเลือกใช้คำสั่งการชดเชยขนาตตามเส้นขอบรูปมีดังนี้

G41 ใช้เมื่อมีดกัดอยู่ทางด้านซ้ายของขอบงาน เมื่อมองตามทิศทางการเคลื่อนที่เดินกัดของมีดกัด

G42 ใช้เมื่อมีดกัดอยู่ทางด้านขวาของขอบงาน เมื่อมองตามทิศทางการเคลื่อนที่เดินกัดของมีดกัด

ข้อควรระวังในการใช้โปรแกรมการชดเชยขนาดตามเส้นขอบรูปก็คือ การชดเชยขนาดตามเส้นขอบรูปด้วยคำสั่งG41 และ G42 จะใช้ได้เฉพาะกับจุดโคออดิเนทที่อยู่บนระนาบ(Plane) เดียวกันเท่านั้นคือ ระนาบ X/Y หรือ Y/Z หรือ X/Z เช่น เมื่อเขียนโปรแกรมการทำงานอยู่บนระนาบของ X/Y การชดเชยขนาดจะกระทำได้เฉพาะแกน X และ Y เท่านั้น ส่วนในแกน Zที่ใช้เป็นแนวแกนในการป้อนความลึก จะไม่สามารถใช้ค่าชดเชยขนาดจากคำสั่งนี้ได้จะต้องแยกการชดเชยขนาดไว้ในขั้นตอนก่อนหน้าที่จะมีการชดเชยขนาดตามเส้นขอบรูป

คำสั่ง G40, G41 และ G42 จะเป็นคำสั่งที่คงอยู่ในโปรแกรมตลอด หลังจากที่ได้โปรแกรมเข้าไปในระบบควบคุมเครื่องจักรกลแล้ว[2]

2.9 องค์ประกอบของคำสั่งG41/G42

เมื่อใช้คำสั่ง G41 และ G42 ระบบควบคุมจะถามหาข้อมูลต่างๆที่จำเป็นต้องในการชดเชยขนาดตามเส้นขอบรูป ข้อมูลต่างๆที่จำเป็นต้องป้อนให้กับระบบควบคุมมีดังนี้[2]

Dg = หมายเลขการชดเชยขนาดมีดกัด (Tool compensation number)

G4 = คำสั่งการเคลื่อนที่เข้าหาชิ้นงาน

A+... = ระยะเคลื่อนที่เข้าหาชิ้นงาน

X... = จุดเริ่มต้นกัดของรูปของชิ้นงานในแนวแกน X

Y... = จุดเริ่มต้นกัดของรูปของชิ้นงานในแนวแกน Y

G0... = ความเร็วในการเคลื่อนที่ไปจุดเริ่มต้นเข้าหาชิ้นงาน

G6... = คำสั่งควบคุมการเคลื่อนที่ตามเส้นขอบรูป

M6... = การควบคุมอัตราป้อนตามเส้นขอบรูป

ความหมายของข้อมูลแต่ละตัวที่กล่าวมาข้างต้น สามารถแยกกล่าวได้ดังนี้

Dg+ : **หมายเลขการชดเชยขนาดมีดกัด**

รัศมีของมีดกัดจะถูกบันทึกเก็บไว้ภายใต้หมายเลขการชดเชยขนาดมีดกัดที่เลือกใช้ รัศมีของมีดกัดจะถูกนำมาใช้ เป็นค่าในการชดเชยขนาดของมีดกัด หมายเลขการชดเชยขนาดมีดกัดจะมีให้เลือกใช้ได้ทั้งหมด 100 หมายเลข คือตั้งแต่หมายเลข D00 – D99 ทำให้สามารถเก็บบันทึกค่าในการชดเชยขนาดมีดกัดได้ถึง 100 ค่า และเรียกออกมาใช้ในการชดเชยขนาดตามเส้นขอบรูปตามหมายเลขการชดเชยขนาด Dg+ ที่ป้อนไว้ในโปรแกรม

G4.. : **คำสั่งการเคลื่อนที่เข้าหาชิ้นงาน**

คำสั่งการเคลื่อนที่เข้าหาชิ้นงาน เป็นคำสั่งที่ใช้ในการกำหนดลักษณะ การเคลื่อนที่เข้ากัดชิ้นงานที่จุดเริ่มต้นของขอบรูป ภายหลังจากที่คำสั่งการชดเชยขนาดมีดกัดตามเส้นขอบรูป ถูกเรียกออกมาใช้งานแล้ว

ลักษณะการเคลื่อนที่เข้าหาชิ้นงานมีให้เลือกใช้ได้ 3 ลักษณะด้วยกัน คือ

- เคลื่อนที่เข้าหาชิ้นงานแบบขนานกับเส้นขอบรูป
- เคลื่อนที่เข้าหาชิ้นงานแบบครึ่งวงกลม
- เคลื่อนที่เข้าหาชิ้นงานแบบ ¼ ของวงกลม

การใช้คำสั่งการเคลื่อนที่เข้าหาชิ้นงาน ทำให้สามารถเขียนโปรแกรมการเคลื่อนที่ของมีดกัดเข้ากัดชิ้นงานที่มีขอบรูปเป็นเส้นตรงหรือเส้นโค้งได้ง่าย และช่วยป้องกันไม่ให้เกิดรอยมีดกัดบนผิวงาน

G45 : การเคลื่อนที่เข้าหาชิ้นงานแบบขนานกับเส้นขอบรูป

เมื่อใช้คำสั่ง G45 มีดกัดจะเคลื่อนที่เร็วหรือเคลื่อนที่ตามค่าอัตราป้อนไปยังจุดเริ่มเข้าหาชิ้นงานที่จุดนี้จะเริ่มป้อนกัดลึกเข้าไปในผิวงานของชิ้นงานโดยอัตโนมัติ

การเคลื่อนที่เร็วจะใช้คำสั่งG00 และการเคลื่อนที่ตามค่าอัตราป้อน (F) ใช้คำสั่งG01 ค่าอัตราป้อนที่โปรแกรมไว้ ระยะห่างระหว่างจุดเริ่มเข้าหาชิ้นงานกับขอบรูปจะเท่ากับรัศมีของมีดกัด ซึ่งก็คือ ค่าชดเชยขนาดมีดกัดที่โปรแกรมไว้ (Dg) นั่นเอง

G46 : การเคลื่อนที่เข้าหาชิ้นงานแบบครึ่งวงกลม

เมื่อใช้คำสั่ง G46 มีดกัดจะเคลื่อนที่ไปยังจุดเริ่มต้นเข้าหาชิ้นงานแบบเคลื่อนที่เร็วหรือตามค่าอัตราป้อนที่โปรแกรมไว้

การเคลื่อนที่เร็วจะใช้คำสั่ง G00และการเคลื่อนที่ตามค่าอัตราป้อนจะใช้คำสั่งG01 เมื่อป้อนกัดลึกจนถึงความลึกที่โปรแกรมไว้แล้ว มีดกัดจะเดินกัดเข้าหาชิ้นงานในแนวครึ่งวงกลมด้วยความเร็วตามอัตราป้อนที่ใช้

ระบบควบคุมจะชดเชยขนาดของรัศมีของมีดกัด ตามค่าที่โปรแกรมไว้ในหมายเลขของการชดเชยขนาด (Dg+)

G47 : การเคลื่อนที่เข้าหาชิ้นงานแบบ 1/4ของวงกลม

เมื่อใช้คำสั่ง G47 มีดกัดจะเคลื่อนที่ไปยังจุดเริ่มต้นเข้าหาชิ้นงาน แบบเคลื่อนที่เร็วเมื่อใช้คำสั่งG00 ประกอบ และจะเคลื่อนที่ตามอัตราป้อนเมื่อใช้คำสั่ง G01 ประกอบในโปรแกรมเมื่อสิ้นสุดการป้อนความลึก มีดกัดจะเคลื่อนที่เข้าหาขอบรูปของชิ้นงานในแนว ¼ ของวงกลมด้วย ความเร็วตามค่าอัตราป้อนที่ใช้ ไปยังจุดเริ่มต้นกัดชิ้นงานที่โปรแกรมไว้ โดยที่ระบบควบคุมจะทำการชดเชยขนาดรัศมีของมีดกัดตามค่าชดเชยขนาดมีดกัดที่โปรแกรมไว้ในหมายเลขการชดเชยขนาด Dg+ โดยอัตโนมัติ

G0.. : ความเร็วในการเคลื่อนที่ไปยังจุดเริ่มต้นเข้าหาชิ้นงาน

คำสั่ง G0..ในโปรแกรมการชดเชยขนาดตามเส้นขอบรูปจะเป็นตัวกำหนดความเร็วในการเคลื่อนที่ไปยังจุดเริ่มต้นเข้าหาชิ้นงาน กล่าวคือ

G00 : เป็นการเคลื่อนที่เร็วไปยังจุดเริ่มต้นเข้าหาชิ้นงาน

G01 : เป็นการเคลื่อนที่ตามค่าอัตราป้อนไปยังจุดเริ่มต้นเข้าหาชิ้นงาน

G6... : คำสั่งควบคุมการเคลื่อนที่ตามเส้นขอบรูป

คำสั่ง G6.. จะเป็นคำสั่งที่ช่วยควบคุมการเคลื่อนที่ของมีดกัดตรงรอยต่อของขอบรูป โดยเฉพาะตรงมุมด้านในของขอบรูป คำสั่งควบคุมการเคลื่อนที่ของมีดกัดตามเส้นขอบรูปมีให้เลือกใช้ได้ 3 ลักษณะคือ

G60 : หยุดเคลื่อนที่ชั่วคราว

เมื่อใช้คำสั่งG60 ประกอบในคำสั่งการชดเชยขนาดมีดกัดตามเส้นขอบรูป เมื่อมีดกัดเคลื่อนที่กัดขอบรูปตรงมุมด้านใน มีดกัดจะเคลื่อนที่ไปยังจุดศูนย์กลางของรัศมีมุมด้านในและหยุดชั่วขณะหนึ่งแล้วจึงเคลื่อนที่เดินกัดต่อไป ดังนั้น คำสั่งG60 จะใช้สำหรับงานกัดมุมด้านในของขอบรูปให้ขนาดสำเร็จและมีผิวงานเรียบ

G61 : เคลื่อนที่กับรัศมีมุมด้านในโดยอัตโนมัติ

เมื่อคำสั่ง G61 มีดกัดจะเคลื่อนที่เดินกัดเป็นรัศมีตรงรอยต่อของมุมด้านใน โดยเคลื่อนที่เป็นรัศมีที่โตกว่ารัศมีที่ต้องการโดยอัตโนมัติ เพื่อป้องกันไม่ให้มีดกัดดูดเข้าไปตรงมุมของชิ้นงาน ซึ่งจะทำให้ขนาดรัศมีที่ได้โตเกินไปหรือเสียขนาดไป สำหรับมีดกัดขนาดรัศมี 4 มม. และเล็กกว่ารัศมีการเคลื่อนที่จะเท่ากับ 0.4 มม. ส่วนมีดกัดที่มีขนาดรัศมีโตกว่า 4 มม. รัศมีการเคลื่อนที่จะเท่ากับ 10% ของรัศมีมีดกัด

การเคลื่อนที่ของมีดกัดลักษณะเช่นนี้ จะทำให้รัศมีรอยกัดที่ขอบรูปของชิ้นงานมีขนาดรัศมีโตกว่าค่าที่โปรแกรมไว้ ดังนั้น คำสั่ง G61 จึงใช้สำหรับงานกัดหยาบของขอบรูปส่วนที่เป็นรัศมีด้านในเพื่อป้องกันไม่ให้ขอบรูปเสียขนาด

G64 : เคลื่อนที่กัดต่อเนื่องสม่ำเสมอตลอดรอยต่อของขอบรูป

เมื่อต้องการให้มีดกัดเคลื่อนที่อย่างต่อเนื่องผ่านรอยต่อของขอบรูป เช่น งานกัดขอบรูปที่เป็นส่วนโค้งต่อกัน เป็นต้น ควรใช้คำสั่ง G64 เพราะการเคลื่อนที่ของมีดกัดจะไม่หยุดระหว่างรอยต่อของแต่ละบล็อก

M6.. : การควบคุมอัตราป้อนตามเส้นขอบรูป

ในขณะที่มีดกัดเคลื่อนที่เดินกัดขอบรูปของชิ้นงานที่มีรัศมีโค้งภายนอกและภายใน จะทำให้ความเร็วในการเคลื่อนที่เดินกัดที่รัศมีด้านในและด้านนอกต่างกัน ซึ่งจะมีผลทำให้คุณภาพของความเร็วในการเคลื่อนที่เดินกัดที่รัศมีด้านในและด้านนอกต่างกัน ซึ่งมีผลทำให้คุณภาพของผิวงานที่ได้แตกต่างกันด้วย ดังนั้น ในคำสั่งการชดเชยขนาดมีดกัดตามเส้นขอบรูป จึงได้จัดเตรียมคำสั่งในการควบคุมอัตราป้อนในการเคลื่อนที่ของมีดกัดไว้ ซึ่งมีให้เลือกใช้ได้ 3 ลักษณะดังนี้

M60 : อัตราการป้อนที่ขอบคุมตัดคงที่ตลอดเส้นขอบรูป

ในงานกัดที่ต้องการอัตราป้อนที่คงที่ตลอดตรงตำแหน่งขอบคมตัด ดังนั้น อัตราป้อนที่รัศมีด้านในจะต้องลดลงและอัตราป้อนที่รัศมีด้านนอกจะต้องเพิ่มขึ้น ลักษณะเช่นนี้จะต้องใช้คำสั่ง M60 ซึ่งเหมาะสำหรับงานกัดผิวสำเร็จที่ปลายด้านหน้าของคมมีดไม่สัมผัสหรือกัดชิ้นงาน คือเดินกัดชิ้นงานด้วยคมด้านข้างของมีดกัดเท่านั้น

M61 : อัตราป้อนที่ขอบคมตัดคงที่ตลอดเส้นขอบรูป (โดยไม่เพิ่มอัตราป้อนที่รัศมีด้านนอก)

เมื่อใช้คำสั่งM61 อัตราป้อนที่ขอบคมตัดขณะเดินกัดรัศมีด้านใน จะมีค่าคงที่ตลอดเช่นเดียวกับการใช้คำสั่ง M60 แต่เมื่อเดินกัดรัศมีด้านนอก อัตราป้อนจะไม่เพิ่มขึ้นและมีค่าคงที่ตลอดซึ่งหมายความว่า อัตราป้อนที่ขอบคมตัดเมื่อเดินกัดรัศมีด้านนอกจะลดลง

M62 : อัตราป้อนที่แนวแกนของมีดกัดคงที่

เมื่อใช้คำสั่ง M62 อัตราป้อนที่แนวแกนหรือศูนย์กลางของมีดกัดจะมีค่าคงที่ตลอด ไม่ว่าจะเป็นงานกัดรัศมีด้านนอกหรือด้านในก็ตาม

การยกเลิกคำสั่งการชดเชยขนาดตามเส้นขอบรูป : G40

คำสั่งการชดเชยขนาดมีดกัดตามเส้นขอบรูป G41/G42 เป็นคำสั่งที่มีผลต่อเนื่องถึงบล็อกอื่นๆที่เขียนตามหลังบล็อกที่ใช้คำสั่งการชดเชยขนาดมีดกัด ดังนั้น เมื่อสิ้นสุดการกัดชิ้นงานตามเส้นขอบรูปแล้ จะต้องยกเลิกการใช้คำสั่งนี้ก่อนที่จะเลื่อนมีดกัดไปยังตำแหน่งอื่นๆ คำสั่งที่ใช้ในการยกเลิก

การใช้คำสั่งการชดเชยขนาดมีดกัดตามเส้นขอบรูป คือ G40 ซึ่งประกอบไปด้วยคำสั่งการเคลื่อนที่ออกจากขอบรูปของชิ้นงานและระยะเลื่อนออกด้วย

2.10 โปรแกรมเมเบิล ลอจิก คอนโทรลเลอร์ (PLC)

PLC เป็นอุปกรณ์ซนิด โซลิด – เสตท (Solid State) ที่ทำงานแบบลอจิก (Logic Function) การออกแบบการทำงานของ PLC จะคล้ายกับหลักการทำงานของคอมพิวเตอร์ จากหลักการพื้นฐานแล้ว PLC จะประกอบด้วยอุปกรณ์ที่เรียกว่า Solid – State Digital Logic Elements เพื่อทำงานและตัดสินใจแบบลอจิก PLC ใช้สำหรับควบคุมกระบวนการทำงานของเครื่องจักรและอุปกรณ์ในโรงงานอุตสาหกรรม

การใช้ PLC สำหรับควบคุมเครื่องจักรหรืออุปกรณ์ต่างๆ ในโรงงานอุตสาหกรรม จะมีข้อได้เปรียบการใช้ระบบของรีเลย์ (Relay) ซึ่งจำเป็นต้องเดินสายไฟฟ้าหรือที่เรียกว่า Hard-Wired ฉะนั้นเมื่อมีความจำเป็นที่ต้องเปลี่ยนกระบวนการผลิต หรือลำดับการทำงานใหม่ต้องเดินสายไฟฟ้าใหม่ ซึ่งเสียเวลาและเสียค่าใช้จ่ายสูง แต่เมื่อเปลี่ยนมาใช้ PLC แล้วการเปลี่ยนกระบวนการการผลิตหรือลำดับในการทำงานใหม่นั้น ทำได้โดยการเปลี่ยนโปรแกรมใหม่เท่านั้น นอกจากนี้แล้ว PLC ใช้ระบบโซลิด – สเตท ซึ่งน่าเชื่อถือว่าระบบเดิม การกินกระแสไฟฟ้าน้อยกว่าและสะดวกกว่าเมื่อต้องการขยายขั้นตอนการทำงานของเครื่องจักร[3]

การทำงานของ PLC

เครื่องจักรที่ควบคุมด้วย PLC จะมีความสามารถเขียนโปรแกรมการทำงานของเครื่องจักรแล้วมีความยืดหยุ่นในการเขียนโปรแกรม เช่น การเปลี่ยนแปลงแก้ไขเพิ่มเติมก็สามารถกระทำได้ซึ่งรวมถึงไทเมอร์ เคาน์เตอร์หรือคำสั่งพิเศษต่างๆ เช่น MOV Data และอื่นๆ อีกมากมาย เพื่อใช้ควบคุมอุปกรณ์ภายนอก ไม่ว่าจะเป็นมอเตอร์ โซลีนอยด์ หลอดไฟ เป็นต้น นอกจากนี้ยังมีการติดต่อสื่อสารกับ PLC กับคอมพิวเตอร์ เพื่อแลกเปลี่ยนข้อมูลระหว่างกันหรืออาจจะติดต่อกับจอซนิดสัมผัสเพื่ออำนวยความสะดวกต่อสัญญา INPUT-OUTPUT ยิ่งกว่านั้นการติดต่อกับคอมพิวเตอร์เพื่อให้คอมพิวเตอร์เป็นตัวควบคุม PLC อีกทีหนึ่ง ซึ่งทำให้ขีดความสามารถควบคุมมอเตอร์กวนของเหลวในภาชนะโดยมีเงื่อนไขต่างๆ

มอเตอร์ไฟฟ้าจะหมุนได้เมื่อหน้าคอนแทคของเซนเซอร์อุณหภูมิและเซนเซอร์ความดันต่อกัน การต่อวงจรทำได้จากการใช้วงจรรีเลย์และการใช้ PLC ในกรณีของการใช้วงจรรีเลย์นั้นมอเตอร์

ทำงานได้เมื่อสวิตช์ของอุณหภูมิและความดันต่อกันหรือใช้สวิตช์มือกดถูกกด ซึ่งต่อเข้าไปยังขั้วสัญญาณเข้าเป็นการต่ออุปกรณ์ทำงานเข้ากับขั้วสัญญาณออก[3]

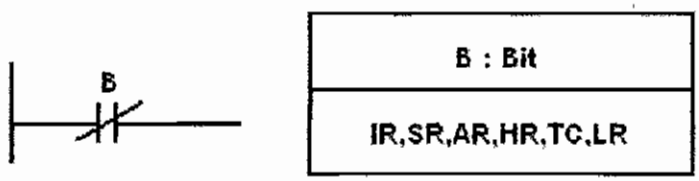
2.11 คำสั่งพื้นฐานของ PLC (OMRON)

คำสั่งเหล่านี้ เป็นคำสั่งที่ใช้เขียนลงในโปรแกรม เพื่อสั่งงานแบบง่ายๆ ทั้งในรูปของแลดเดอร์ และนีมอนิก[3]

2.11.1 **คำสั่ง** LOAD-LD

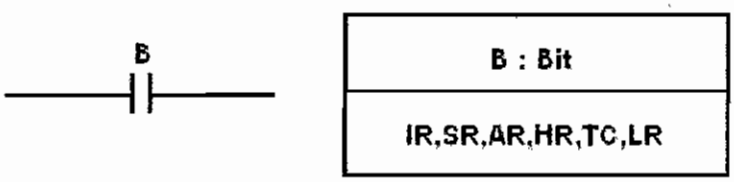
รูปที่ 2.11 คำสั่ง LOAD – LD

2.11.2 **คำสั่ง** LOAD NOT – LD NOT



รูปที่ 2.12 คำสั่ง LOAD NOT – LD NOT

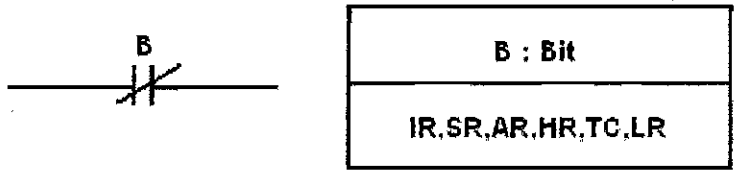
2.11.3 **คำสั่ง** AND - AND



รูปที่ 2.13 คำสั่ง AND – AND

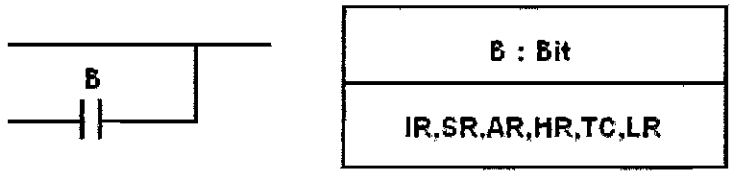
15 มิ.ค. 2549
4840515

2.11.3 คำสั่ง AND NOT – AND NOT



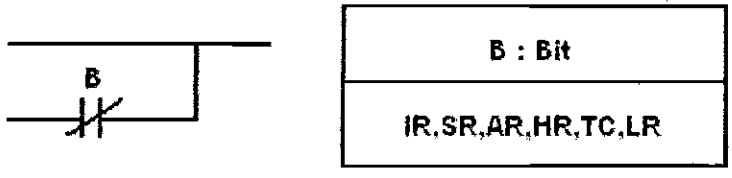
รูปที่ 2.14 คำสั่ง AND NOT – AND NOT

2.11.4 คำสั่ง OR – OR



รูปที่ 2.15 คำสั่ง OR-OR

2.11.5 คำสั่ง OR NOT – OR NOT



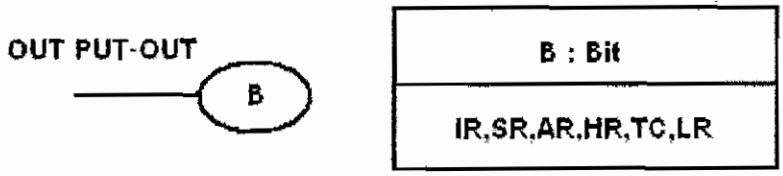
รูปที่ 2.16 คำสั่ง OR NOT – OR NOT

2.11.7 คำสั่ง OR LOAD – OR LD

เป็นคำสั่งใช้รวมบล็อกเข้าด้วยกัน ซึ่งการใช้คำสั่งเพียง AND หรือ OR ไม่สามารถใช้ทำงานตามต้องการได้

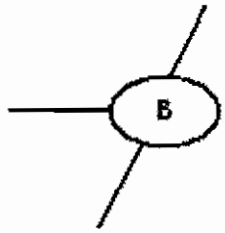
2.11.8 **คำสั่ง** OUTPUT **และ** OUTPUT NOT

เป็นคำสั่งที่สั่งให้ OUTPUT ภายนอกทำงานหรือไม่ทำงาน ถ้า B : ถูกกำหนดเป็น IR แต่ถ้า B : ถูกกำหนดเป็นอย่างอื่นจะเป็นเจ้าของรีเลย์ภายใน



รูปที่ 2.17 คำสั่ง OUTPUT AND OUTPUT NOT

2.11.9 **คำสั่ง** OUTPUT NOT – OUT NOT



รูปที่ 2.18 คำสั่ง OUTPUT NOT – OUT NOT

การทำงานของคำสั่งนี้ จะตรงกันข้ามกับคำสั่ง OUT

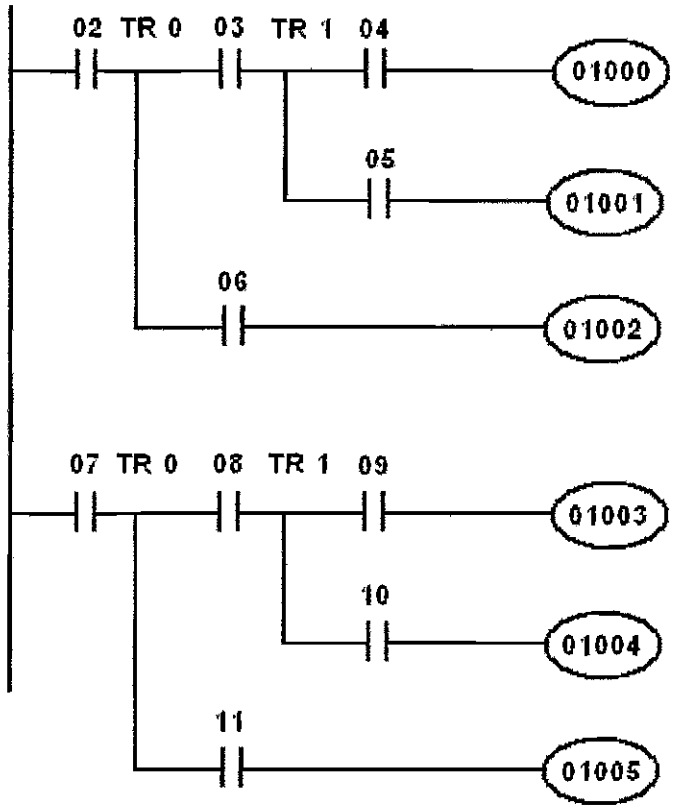
2.11.10 **คำสั่ง** END (FUN 01)

การเขียนโปรแกรมทุกครั้ง เมื่อสิ้นสุดเขียนโปรแกรมแล้วจะต้องจบด้วยคำสั่ง END ถ้าไม่มีคำสั่ง END (FUN 01) เมื่อให้โปรแกรมทำงานจะมีข้อความ NO END INSTR แสดงขึ้น

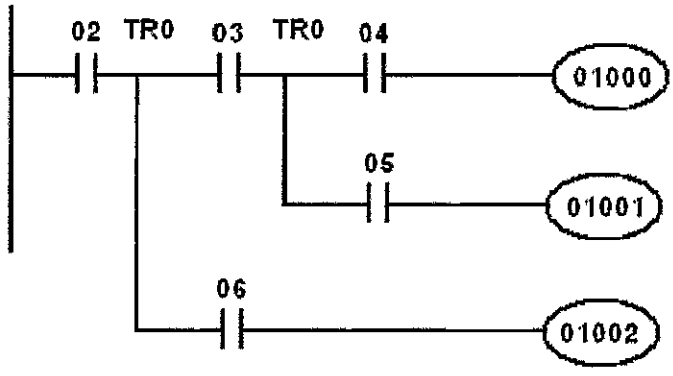
ถ้าไม่ได้ใส่คำสั่ง END (FUN 01) หลังจากป้อนโปรแกรมจบแล้วที่แป้นกดโปรแกรมจะแสดงข้อความ NO END INSTR แสดงว่าไม่มีคำสั่ง END โปรแกรมจะไม่สามารถทำงานได้และหลอดไฟแสดงการทำงานผิดพลาดจะสว่างขึ้น

2.11.11 คำสั่ง TR (Temporary Memory Relay)

คำสั่งนี้ใช้กับแลดเดอร์ที่มีเอาต์พุต คอยล์ อยู่หลายสาขาโดยที่สาขาหนึ่งๆ ประกอบไปด้วยคำสั่งTR นี้หลายตัว และคำสั่ง TR มีให้เรียกใช้ตั้งแต่ TR0 จนถึง TR7 ซึ่งในสาขาเดียวจะใช้ TR ซ้ำกันไม่ได้ แต่ถ้าเป็นสาขาใหญ่หลายสาขาจะได้ TR0 – TR7 ในสาขาใหญ่นั้นได้อีก



แลดเดอร์ไดอะแกรมที่ใช้งานได้



แลดเดอร์ไดอะแกรมที่ใช้งานไม่ได้

รูปที่ 2.19 คำสั่ง TR (Temporary Memory Relay)

2.11.12 **คำสั่ง** Interlock and Interlock Clear – IL (02) and ILC (03)

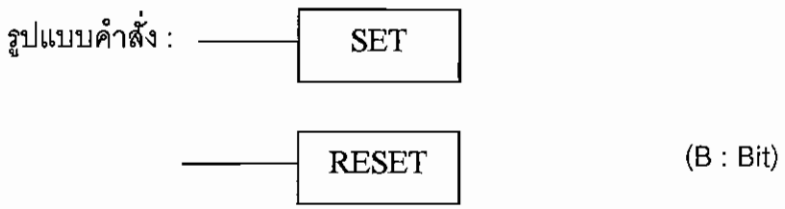
คำสั่ง IL และ ILC จะต้องใช้ร่วมกันคือ ถ้าเริ่มต้นมีการใช้คำสั่งด้วย IL เมื่อใดแล้วถ้าต้องการสิ้นสุดการทำงานต้องจบด้วย ILC เงื่อนไขของคำสั่งนั้น คอนแทคตรงหน้าส่วนของ IL สภาวะ "ON" จะทำให้โปรแกรมที่อยู่ระหว่าง IL และ ILC ทำงานเป็นปกติ แต่ถ้าคอนแทคตำแหน่งดังกล่าวมีสภาวะ "OFF" จะทำให้การทำงานของโปรแกรมระหว่าง IL และ ILC ไม่ทำงาน ในขณะเดียวกันสัญญาณเอาต์พุต ในช่วงนั้นจะมีสภาวะ "OFF" ด้วย

2.11.13 **คำสั่ง** JMP (FUN 04) **และ** LME (FUN 05)

การใช้งานของคำสั่งนี้จะต้องใช้งานคู่กัน เงื่อนไขต่างๆ ที่อยู่ระหว่างคำสั่ง JMP และ JME จะมีเงื่อนไขการทำงานเป็นปกติ ในกรณีที่ชุดของคอนแทคตรงส่วนหน้าของ JMP มีสภาวะเป็น "ON" แต่ถ้าชุดคอนแทคดังกล่าวมีสภาวะเป็น "OFF" เมื่อใด OUTPUT, TIMER, COUNTER, KEEP อยู่ระหว่างคำสั่งชุดคอนแทคดังกล่าวเอาไว้เช่นเดิม และจะมีการเปลี่ยนแปลงอีกครั้ง ถ้าชุดคอนแทคมีสภาวะ "ON" สามารถใช้ JUMP 00 ได้หลายครั้งตามต้องการ แต่ JUMP 00 ถึง 99 สามารถใช้ได้เพียงครั้งเดียว

2.11.14 **คำสั่ง** SET **และ** RESET-SET-RSET

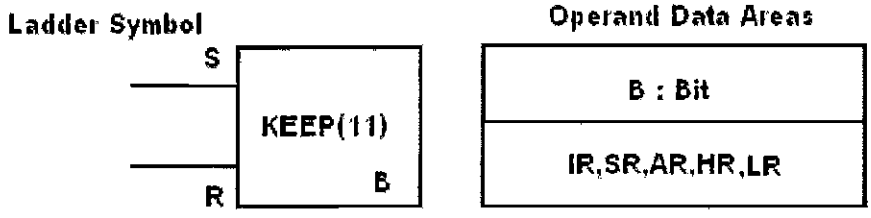
เป็นคำสั่งเมื่อมีสภาวะ "ON" แล้วจะยังคงค้างสภาวะ "ON" อยู่จนกว่าคำสั่ง RESET ที่ BIT เดียวกัน มีสภาวะ "ON" ถึงแม้ว่าอินพุตเลิกทำงานไปแล้วก็ตาม



รูปที่ 2.20 คำสั่ง SET และ RESET – SET – RSET

2.11.15 **คำสั่ง** KEEP – KEEP (11)

เหมือนกับคำสั่ง SET และ RESET เพียงแต่รวม SET และ RESET ให้อยู่ในตัวเดียวกัน เพื่อให้ผู้ใช้งานสามารถเลือกใช้โปรแกรมได้อย่างสะดวกตามความเหมาะสม

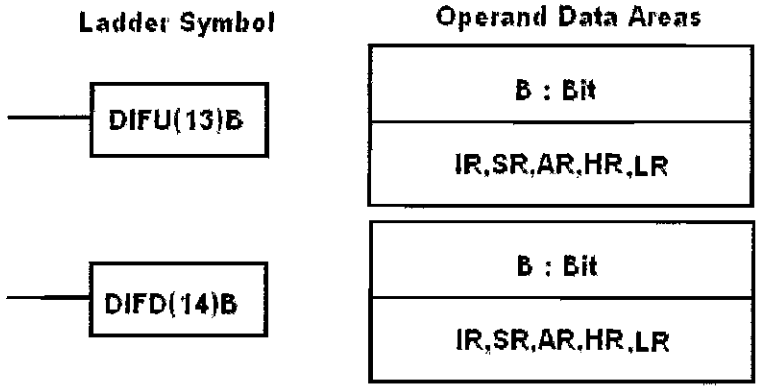


รูปที่ 2.21 คำสั่ง KEEP – KEEP

เมื่อ S ถูก SET "ON" ทำให้บิตที่B ทำงานตลอดไปจนกว่าR จะถูก SET "ON" จึงทำให้บิด B เลิกทำงาน

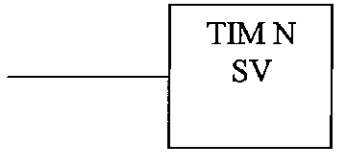
2.11.16 **คำสั่ง** DIFFERENTIATE UP and DOWN – DIFU (13), DIFD (14)

คำสั่ง DIFU (13) และ DIFD (14) เป็นคำสั่งที่ทำงานเพียงขอบขาขึ้นหรือขอบขาลงมาอินพุดเท่านั้น และทำงานเพียงช่วง One Cycle Time เท่านั้น



รูปที่ 2.22 คำสั่ง DIFFERENTIATE UP and DOWN – DIFU (13), DIFD (14)

2.11.17 **คำสั่ง** Timer – TIM



รูปที่ 2.23 คำสั่ง Timer – TIM

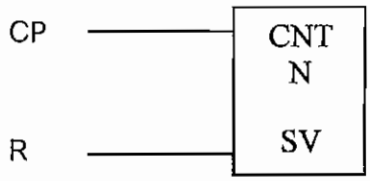
N = จำนวนไทเมอร์ (ใช้ได้ตั้งแต่หมายเลข 000 ถึง 127)

SV = Set Value (สามารถตั้งค่าเวลาได้ทั้ง BCD หรือเป็น Word ก็ได้) และสามารถตั้งค่าได้ตั้งแต่ 0 - 999.9 วินาที

เมื่ออินพุตมีสภาวะเป็น ON คำสั่ง TIM นี้จะเริ่มนับเวลาตามค่าที่ตั้งไว้ในไทเมอร์ เมื่อนับครบตามเวลาที่ตั้งไว้แล้ว เอาต์พุตของไทเมอร์ก็จะ ON แต่ถ้าอินพุตมีสภาวะเป็น OOF ก่อนที่ค่าเวลาของไทเมอร์จะนับถึง ค่าที่ถูกนับไว้ของไทยเมอร์ในขณะนั้นจะถูกตั้งค่าใหม่เป็นศูนย์ทันทีสามารถตั้งเวลาการนับได้ 0.1 วินาที ต่ำสุด

2.11.18 คำสั่ง Counter – CNT

เป็นคำสั่งที่ใช้นับจำนวนครั้งของสัญญาณอินพุตที่สภาวะ "ON" ในแต่ละครั้งและการนับค่านั้น จะนับลงจากที่ตั้งเอาไว้



รูปที่ 2.24 คำสั่ง Counter – CNT

N = จำนวนหมายเลขของเคาน์เตอร์

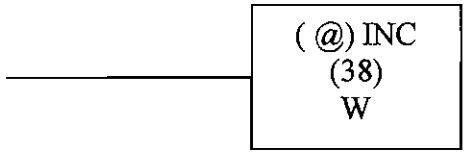
SV = เป็น Set Value สามารถตั้งค่านับได้ 9999 ครั้ง หรือจะตั้งค่าเป็น Word ก็ได้

การทำงานคือ เมื่อขา CP "ON" ครบจำนวนครั้งตามที่ตั้งค่า (SET) ในเคาน์เตอร์ ทำให้เอาต์พุตของ CNT "ON" และจะทำงานไปจนกว่าขา R ของคำสั่ง CNT "ON" จึงทำให้เอาต์พุตของ CNT เลิกทำงาน

หมายเหตุ

พื้นที่ความจำ (Memory Area) ของไทเมอร์และเคารน์เตอร์ใช้พื้นที่เดียวกัน จึงใช้คำสั่งทั้งไทเมอร์และเคาน์เตอร์กับพื้นที่เดียวกันไม่ได้ ตัวอย่างเช่น ใช้คำสั่งไทเมอร์ นับเบอร์ 0 แล้ว จะใช้คำสั่งเคาน์เตอร์ ที่นับเบอร์ 0 อีกไม่ได้ ต้องใช้นับเบอร์ 1 เป็นต้น

2.11.19 **คำสั่ง** BCD Increment – INC (38)



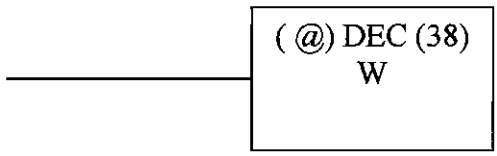
รูปที่ 2.25 คำสั่ง BCD Increment – INC (38)

W = Increment Word (BCD Value)

เมื่อคำสั่งนี้ทำงานค่า Data ใน W จะเพิ่มทีละหนึ่ง (บวกเข้าไปทีละหนึ่ง)

2.11.20 **คำสั่ง** BCD Decrement – DEC (39)

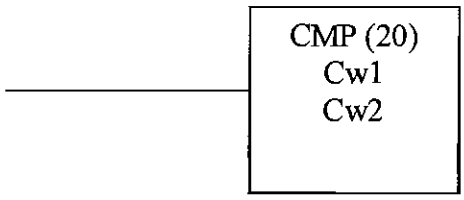
เหมือนกับคำสั่ง INC (38) แต่ตรงกันข้ามคือ ลดทีละ 1



W = Decrement Word (BCD Value)

รูปที่ 2.26 คำสั่ง BCD Decrement –DEC (39)

2.11.21 **คำสั่ง Compare – CMP (20)**



รูปที่ 2.27 คำสั่ง Compare – CMP (20)

CW1 = First Compare Word (หรือค่าคงที่)

CW2 = Second Compare Word (หรือค่าคงที่)

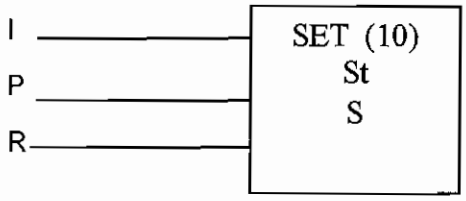
เมื่อคำสั่งนี้ทำงาน ค่าของ CW1 และ CW2 จะถูกเปรียบเทียบกับโดยมีสภาวะ (Condition) ทำงานดังต่อไปนี้

GR Flag (SR 25505) จะ "ON" ถ้า CW1 > CW2

EQ Flag (SR 25506) จะ "ON" ถ้า CW1 = CW2

LE Flag (SR 25507) จะ "ON" ถ้า CW1 < CW2

2.11.22 คำสั่ง Shift Register – SFT (10)



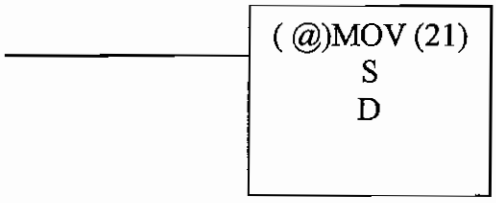
รูปที่ 2.28 คำสั่ง Shift Register – SFT (10)

ST = Strating Word

E = End Word

ถ้าให้ P มีสภาวะเป็น ON แต่ละครั้งการทำงานจะเลื่อน Data ทีละบิตจากบิต 0 ของ Strating Word ไปจนถึงบิต 15 ของEnd Word

2.11.23 คำสั่ง Move – MOV (21)



รูปที่ 2.29 คำสั่ง Move – MOV (21)

S = Source Word (หรือค่าคงที่)

D = DestinationWord (เวิร์ดที่ถูกย้าย)

เมื่อคำสั่งนี้ทำงาน MOV (21) จะย้ายข้อมูลจาก S ยังมีข้อมูลเดิมอยู่ สัญลักษณ์ @ จะทำงาน 1 Scan time