บทที่ 2
หลักการ เหตุผล และทฤษฎีที่เกี่ยวข้อง

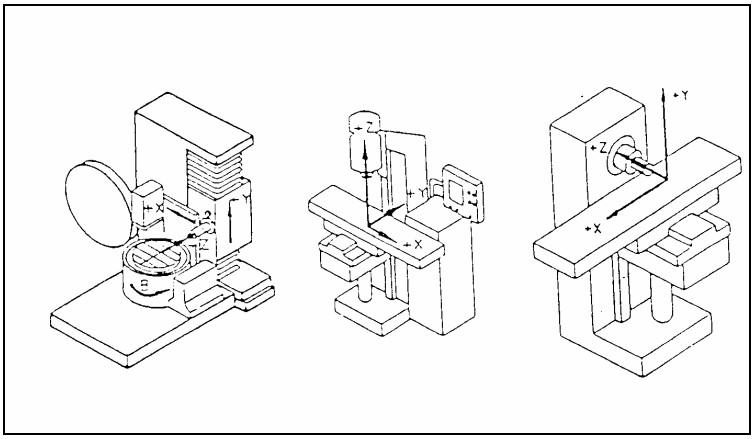
เครื่องกัดแบบซีเอ็นซี มีส่วนประกอบที่สำคัญคือ เครื่องกัดที่มีเสต็ปปิ้งมอเตอร์เป็นส่วนขับเคลื่อน และระบบควบคุมการทำงานให้เกิดความสัมพันธ์กัน ซึ่งในบทนี้จะได้กล่าวถึงทฤษฎีที่เกี่ยวข้องกับส่วนของการทำงานและการควบคุมดังต่อไปนี้

2.1 เครื่องกัด

งานกัด คือการใช้เครื่องมือตัด (Cutting tool) ซึ่งมีความแหลมคมและแข็งกว่าวัสดุงาน เข้าทำการปาดผิววัสดุงานโดยการหมุนเครื่องมือตัดในลักษณะงานผิวราบ ผิวโค้ง และเจาะ การตัดปาดผิวจะเกิดเป็นช่วงเมื่อคมตัดของมีดกัดเข้าตัดเฉือนชิ้นงาน เศษที่เกิดขึ้นจากงานตัดจะขึ้นอยู่กับความสัมพันธ์ของการเคลื่อนที่ของหัวกัดและชิ้นงานโดยการเคลื่อนที่สัมพันธ์จะมีอย่างน้อย 3 ทิศทาง คือ แนวแกน X , Y และ Z ดังรูป 2.1

เครื่องกัดมีหลายชนิด แบ่งตามการทำงานดังนี้

- เครื่องกัดนอน
- เครื่องกัดตั้ง
- เครื่องกัดยูนิเวอร์แซล
- เครื่องกัดพิเศษ เช่น เครื่องกัดเกลียว, เครื่องกัดลอกแบบ



รูป 2.1 ทิศทางของการเคลื่อนที่ในแนวแกน X, Y และZ สำหรับเครื่องกัดแบบต่าง ๆ

การตัดปาดผิว (Machining) เป็นกระบวนการทางการผลิตเกี่ยวกับขนาด, รูปร่าง หรือการเปลี่ยนแปลงคุณสมบัติงานของชิ้นส่วน โดยการนำเอาวัสดุส่วนเกินนั้นออก ดังนั้นการปาดผิวเมื่อเทียบกับการทำงานของเครื่องจักรกลอื่น ๆ จึงเป็นกระบวนการที่มีค่าใช้จ่ายสูงโดยเฉพาะเมื่อต้องการควบคุมความถูกต้องแม่นยำ (Acuracy) สูงและให้ได้ผิวงานสำเร็จที่ดี

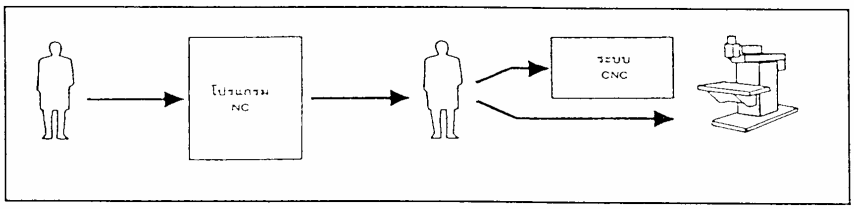
2.2 ระบบซีเอ็นซี

ซีเอ็นซี (CNC: Computer Numerical Control) หมายถึงระบบควบคุมเชิงตัวเลขด้วยคอมพิวเตอร์ ระบบควบคุมนี้นำไปติดตั้งกับเครื่องจักรกล ทำให้สามารถผลิตชิ้นงานทางเรขาคณิตที่ซับซ้อนได้สะดวกรวดเร็วยิ่งขึ้นและมีความเที่ยงตรงสูง เหมาะสำหรับใช้ในงานอุตสาหกรรมการผลิต ระบบซีเอ็นซีเป็นระบบที่พัฒนาขึ้นมาจากระบบ NC (Numerical Control) ระบบควบคุมนี้จะเพิ่มขีดความสามารถในการทำงานของเครื่องจักรกลให้มีประสิทธิภาพเพิ่มมากขึ้น

เครื่องจักรกลซีเอ็นซีจะประกอบด้วยองค์ประกอบใหญ่ 2 ส่วนด้วยกัน คือ

1. เครื่องจักรกล เป็นส่วนที่ทำหน้าที่ตัดเฉือนชิ้นงานตามขั้นตอนการทำงานที่กำหนดไว้

2. ระบบซีเอ็นซี เป็นส่วนทำหน้าที่ควบคุมขั้นตอนการตัดเฉือนทั้งหมด ดังรูป 2.2



รูป 2.2 องค์ประกอบของเครื่องจักรกลซีเอ็นซี

การทำงานของระบบ เอ็นซี และซีเอ็นซี แบ่งออกเป็นส่วนสำคัญดังนี้

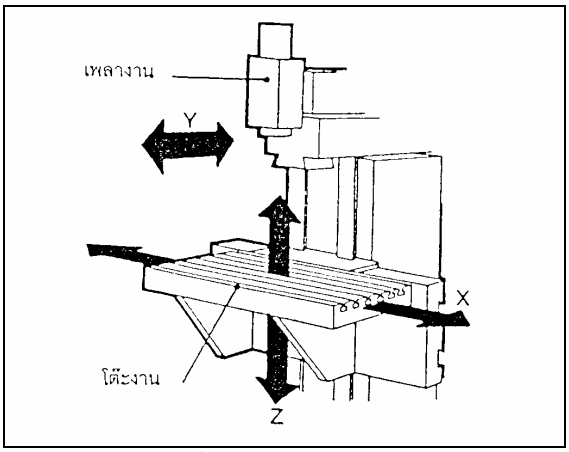
2.2.1 แนวแกนป้อน (Feed Axes)

แนวแกนป้อน หมายถึง แนวการเคลื่อนที่ของหัวกัด หรือชิ้นงาน ซึ่งทำให้เกิดการตัดเฉือนของชิ้นงานให้เป็นรูปร่างตามต้องการ โดยทั่ว ๆ ไป แนวแกนของเครื่องกัดจะมีอยู่ 3 แกน ซึ่งใช้อักษรย่อว่า X, Y และ Z โดยที่ X และ Z จะเป็นการเคลื่อนที่ของโต๊ะงาน ส่วนแกน Y จะเป็นการเคลื่อนที่ของชุดเพลาหัวเครื่อง (ในกรณีที่เครื่องกัดเป็นแบบโต๊ะงานเคลื่อนที่) ดังรูป 2.3

แนวแกนทั้ง 3 แนวแกนที่แสดงในรูปจะทำให้เกิดการเคลื่อนที่ต่าง ๆ ดังนี้

แนวแกน X : โต๊ะงานเคลื่อนที่ไปทางซ้ายและขวา

แนวแกน Y : หัวเพลาเครื่องเคลื่อนที่เข้าและออก
แนวแกน Z : โต๊ะงานเคลื่อนที่ขึ้นและลง



รูป 2.3 เครื่องกัดซีเอ็นซีแบบ 3 แกน

2.2.2 การขับป้อน (Feed Drives)

เป็นการเคลื่อนที่ของแท่นเลื่อน หรือโต๊ะงาน ทำให้เกิดการตัดเฉือนของชิ้นงาน ระบบขับป้อนโดยทั่วไปแล้วมักจะใช้มอเตอร์เป็นตัวขับ โดยควบคุมจากวงจรอิเล็กทรอนิกส์ เครื่องจักรซีเอ็นซีสมัยใหม่จะขับป้อนด้วยเซอร์โว (Servo Drives) ซึ่งสามารถปรับความเร็วรอบได้ มอเตอร์ที่ใช้กับระบบขับป้อนมี 3 ชนิด คือ

ก) มอเตอร์กระแสตรง (DC Motors) เป็นมอเตอร์ที่สามารถปรับอัตราการป้อนได้ละเอียดและมีวงจรไม่ซับซ้อน เหมาะสำหรับเครื่องซีเอ็นซีขนาดเล็ก หรือขนาดกลาง ด้วยเหตุที่เป็นมอเตอร์ที่มีแปรงถ่านจึงทำให้ต้องทำความสะอาดบ่อย และต้องเปลี่ยนแปรงถ่านเนื่องจากแปรงถ่านหมด

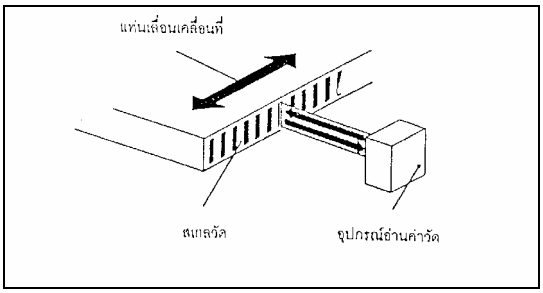
ข) มอเตอร์แบบเป็นขั้น (Stepping Motors) มอเตอร์แบบนี้จะใช้งานประเภทไม่มีสัญญาณป้อนกลับ (Open-loop) ซึ่งจะทำให้ประหยัดค่าใช้จ่าย การออกแบบต้องคำนึงถึงแรงบิด (Torque) ที่จะต้องใช้ด้วย เพราะมอเตอร์แบบนี้จะให้แรงบิดต่ำ ซึ่งเหมาะสมกับเครื่องจักรขนาดเล็ก ๆ

ค) มอเตอร์กระแสสลับ (AC Motors) เป็นมอเตอร์ที่ไม่ต้องใช้แปรงถ่าน ทำให้สามารถลดงานบำรุงรักษาได้มากและให้แรงบิดได้ดีกว่ามอเตอร์กระแสตรง

2.2.3 ระบบวัดตำแหน่ง (Measuring System)

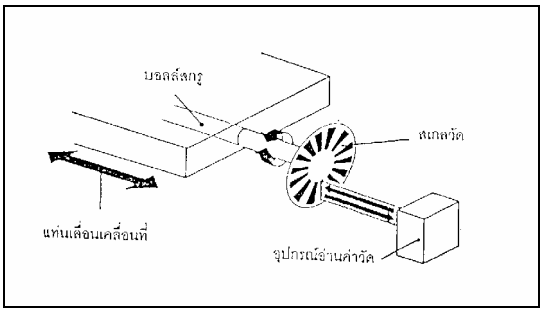
การเคลื่อนที่ไปยังตำแหน่งต่าง ๆ ในแนวแกนของแท่นเลื่อน จะถูกส่งไปยังระบบควบคุมโดยระบบวัดขนาด การวัดตำแหน่งของแท่นสามารถจะวัดได้ทั้งโดยตรง (Direct Measurement) และโดยทางอ้อม (Indirect Measurement)

ก) การวัดตำแหน่งโดยใช้สเกลวัด (Measuring Scale) ยึดติดกับแท่นเลื่อนหรือโต๊ะงานโดยตรง ดังแสดงในรูป 2.4 วิธีวัดแบบนี้ให้ความเที่ยงตรงสูง เพราะอุปกรณ์อ่านค่าวัด (Measuring Valve Resolver) จะอ่านข้อมูลในการวัดจากขีดสเกลวัด (Measuring Scale Grid) และแปลงข้อมูลนี้เป็นสัญญาณไฟฟ้าและส่งกลับไปยังระบบควบคุม



รูป 2.4 การวัดตำแหน่งโดยตรง

ข) การวัดตำแหน่งทางอ้อมวิธีนี้จะใช้สเกลวัดติดกับบอลสกรู หรือมอเตอร์ ซึ่งอุปกรณ์อ่านค่าวัดจะบันทึกการเคลื่อนที่ของการหมุนของแผ่นจานสัญญาณ (Pulse Disc) ที่ติดอยู่กับบอลสกรู และส่งต่อไปยังระบบควบคุมของเครื่อง ระบบควบคุมก็จะใช้สัญญาณที่ได้รับนี้ไปคำนวณหาระยะทางการเคลื่อนที่ของแท่นเลื่อนจานสัญญาณการหมุน (Rotation Pulses) ของจานแผ่นสัญญาณ ดังแสดงในรูป 2.5



รูป 2.5 การวัดตำแหน่งทางอ้อม

2.3 เส้นทางเดินซีเอ็นซี

ตัวกำหนดเส้นทางเดินของหัวกัด คือ อุปกรณ์อิเล็กทรอนิกส์หรือโปรแกรมซอฟแวร์ที่มีหน้าที่ 2 อย่างด้วยกัน คือ การคำนวณความเร็วของหัวกัดตามแนวแกนของแต่ละแกนตามที่ให้อัตราป้อนมา และสร้างพิกัดตามเส้นทางการเดินของโปรแกรม

เส้นทางเดินของหัวกัดสามารถจำแนกได้ 5 แบบ คือ เส้นตรง เส้นโค้ง เกลียว (Helical) พาราโบลาและลูกบาศก์ ซึ่งเครื่องซีเอ็นซีส่วนใหญ่จะใช้เส้นทางเดินแบบเส้นตรงและเส้นโค้ง

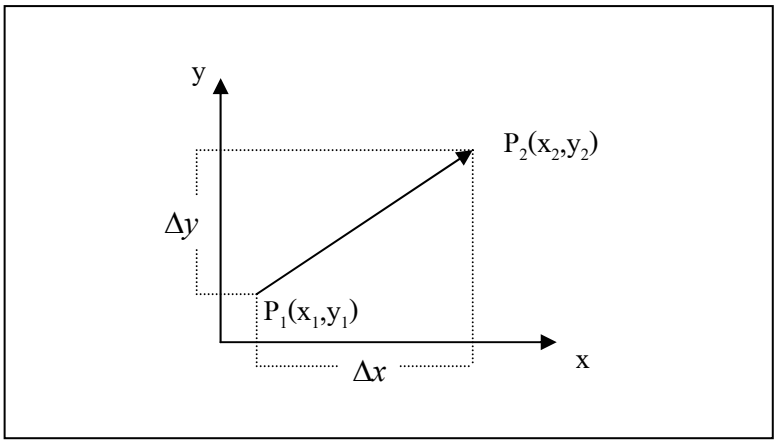
2.3.1 เส้นทางเดินแบบเส้นตรง (Linear Interpolation)

เส้นทางเดินแบบเส้นตรง หัวกัดจะเคลื่อนที่จากจุดเริ่มต้นไปยังจุดสุดท้ายเป็นเส้นตรง ในโปรแกรมของการเคลื่อนที่แบบเส้นตรง จะมีพิกัดของจุดสุดท้ายของแต่ละเส้น เส้นทางเดินแบบเส้นตรงสามารถทำให้เกิดการเคลื่อนที่พร้อมกันของแกนทั้ง 5 ได้ ซึ่งอาจประกอบไปด้วยแกนแนวเส้นตรง 3 แกน (X,Y,Z) และแกนหมุน 2 แกน (A,B) เพื่อจะสร้างเส้นทางของหัวกัดสำหรับรูปร่างต่าง ๆ โดยวิธีทั่ว ๆ ไปที่ใช้จะเป็น 2 – 3 แนวแกนเส้นทางเดินแบบเส้นตรง

เส้นทางเดินแบบเส้นตรงต้องการพารามิเตอร์ 3 ตัว คือ จุดเริ่มต้น จุดสุดท้าย และความเร็ว สำหรับแต่ละแนวแกน ในลักษณะของ 2 แนวแกน ดังรูป 2.6 เส้นทางการเดินแบบเส้นตรงจะคำนวณความเร็วในรูปของพัลซ์ต่อวินาที สำหรับแกน X และ Y สำหรับความเร็วในแนวแกน X และ Y สามารถแสดงได้ดังสมการ

$$V_x = \frac{\Delta x}{(\Delta x^2 + \Delta y^2)^{1/2}} V_f \qquad \text{.....(1)}$$

$$V_y = \frac{\Delta y}{(\Delta x^2 + \Delta y^2)^{1/2}} V_f \qquad \text{.....(2)}$$



รูป 2.6 เส้นเดินทางแบบเส้นตรงจากจุด P_1 ไปยังจุด P_2

2.3.2 เส้นทางเดินแบบวงกลม (Circular Interpolation)

เส้นทางเดินแบบวงกลม ขยายการแตกของเส้นตรงที่ต่อกันเพื่อให้ได้ความละเอียดสูง ตัวกำหนดเส้นทางเดินจะคำนวณความเร็วของ V_x และ V_y และสร้างลำดับของพัลซ์อ้างอิงของแต่ละการควบคุมการเคลื่อนที่ของแกน จำนวนสะสมของพัลซ์จะแสดงตำแหน่งของแกน ส่วนความถี่ของพัลซ์จะแสดงความเร็วของแกน

เส้นทางการเดินแบบวงกลมจะถูกกำหนดจากจุดศูนย์กลาง P_0 และจุดสิ้นสุดของการเคลื่อนที่ 2 จุดคือ P_1 และ P_2 ดังแสดงในรูป 2.7 ตำแหน่งเชิงมุมของหัวกัด (Ø) ซึ่งสามารถวัดได้จากการอ้างอิงจุด P_0 ความเร็วในการป้อนชิ้นงาน V_f สามารถคำนวณได้จาก

$$V_f = R\frac{d\phi}{dt} \qquad \text{.....(3)}$$

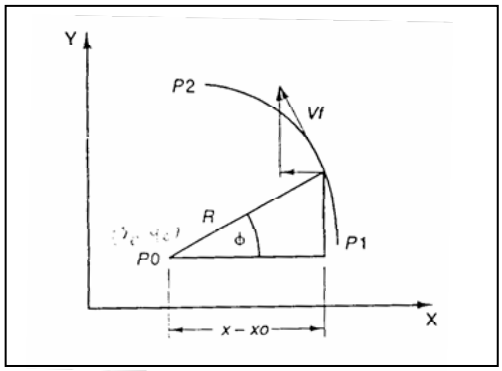
โดยที่ R เป็นรัศมีของเส้นโค้งวงกลม

$$x = R\cos\phi + x_0$$

$$y = R\sin\phi + y_0$$

$$R\cos\phi = x - x_0 \qquad \text{.....(4)}$$

$$R\sin\phi = y - y_0 \qquad \text{.....(5)}$$



รูป 2.7 เส้นทางเดินแบบเส้นโค้งจากจุด P_1 ไปยังจุด P_2

$$\frac{dx}{dt} = -R\sin\phi\frac{d\phi}{dt}$$

$$= -(y - y_0)\frac{d\phi}{dt} \qquad \text{.....(6)}$$

$$\frac{dy}{dt} = R\cos\phi\frac{d\phi}{dt}$$
$$= (x - x_0)\frac{d\phi}{dt} \quad \text{.....(7)}$$

2.4 เสต็ปปิ้งมอเตอร์

เสต็ปปิ้งมอเตอร์ (Stepping Motor) เป็นอุปกรณ์เชิงกลที่ใช้งานกันอย่างแพร่หลาย เช่น Plotter, Disk drive, แขนกลและหุ่นยนต์เป็นต้น เนื่องจากเสต็ปปิ้งมอเตอร์เป็นมอเตอร์ที่สามารถควบคุมตำแหน่งในการหมุนได้ การทำงานของเสต็ปปิ้งมอเตอร์จะทำงานแตกต่างจากมอเตอร์กระแสตรง เพราะเสต็ปปิ้งมอเตอร์จะมีการทำงานแบบลำดับหรือเสต็ป ตามสัญญาณพัลส์ที่ป้อนเข้า

2.4.1ลักษณะการทำงานของเสต็ปปิ้งมอเตอร์

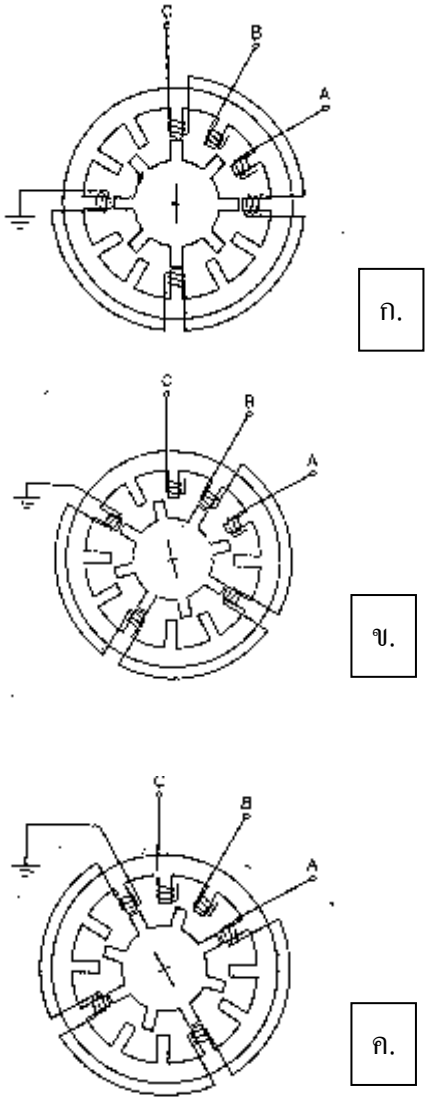
เสต็ปปิ้งมอเตอร์หรือบางครั้งเรียกว่า เสต็ปมอเตอร์ หรือเสต็ปเปอร์มอเตอร์ เป็นอุปกรณ์ขับเคลื่อนทางกลที่ทำงานด้วยไฟฟ้า โดยการขับเคลื่อนกระทำเป็นขั้น มอเตอร์แบบนี้จะเปลี่ยนพลังงานไฟฟ้าเป็นการหมุนทางกล การหมุนของเพลามอเตอร์จะหมุนเป็นขั้น ซึ่งมีขนาดการหมุนเท่ากัน เมื่อมีกระแสไฟฟ้าผ่านคอยล์หลายอันภายในมอเตอร์ โดยการส่งกระแสผ่านคอยล์อันหนึ่งก่อน และตัดกระแสจากคอยล์อันนั้น แล้วจึงจ่ายกระแสให้คอยล์ต่อไปอย่างเป็นลำดับ เสต็ปปิ้งมอเตอร์ที่มีใช้กันอยู่ทั่วไปมักจะเป็นแบบมีคอยล์ภายใน 4 ชุด หรือเรียกว่าเป็นมอเตอร์แบบ 4 เฟส

ลักษณะการขับเคลื่อนแบบนี้จะทำได้โดยไมโครคอมพิวเตอร์หรือไมโครโปรเซสเซอร์ในการควบคุมจ่ายกระแสไฟให้แต่ละเฟส ซึ่งก็จะทำให้สามารถควบคุมทิศทาง ความเร็ว และอัตราเร่งของมอเตอร์ได้ด้วยโปรแกรม การขับมอเตอร์อาจทำได้โดยการขับทีละเฟสในระบบที่ให้กำลังน้อย หรืออาจทำทีละสองเฟสเพื่อให้ได้กำลังมากในบางสถานการณ์ การขับมอเตอร์อาจทำโดยการขับแบบหนึ่งเฟสและสองเฟสสลับกันไป เพื่อให้ได้ความละเอียดของมุมขับสูงเป็นสองเท่า แรงบิดที่ได้จากมอเตอร์ซึ่งจะลดลงเมื่อความเร็วของมอเตอร์เพิ่มขึ้น การออกแบบวงจรขับเคลื่อนมอเตอร์ที่ความเร็วต่ำ สามารถจะกระทำได้ง่าย แต่เมื่อต้องการใช้มอเตอร์ที่ความเร็วสูง วงจรขับจะต้องซับซ้อนขึ้น เพื่อไม่ให้แรงบิดของมอเตอร์ลดลงมากเกินไปเมื่อความเร็วสูงขึ้น ถ้าหากว่ามอเตอร์ใช้ในการขับอุปกรณ์ที่มีความเฉื่อยสูง โปรแกรมในการควบคุมควรจะมีการเร่งและเบรคมอเตอร์อย่างช้า ๆ เพื่อป้องกันไม่ให้เกิดการกระชากและทำให้มอเตอร์ไม่สามารถหมุนไปตามขั้นตอนที่ขับได้

2.4.2 โครงสร้างของเสต็ปปิ้งมอเตอร์

มอเตอร์ที่ใช้กันมากคือ มอเตอร์แบบ Variable Reluctance (VR) ซึ่งจะใช้สเตเตอร์ที่มีคอยล์พันอยู่ เพื่อให้เกิดกระแสแม่เหล็กและตัวโรเตอร์จะไม่มีการกระตุ้นด้วยกระแสไฟฟ้า ในรูป 2.8 แสดงให้เห็นโครงสร้างภายในของมอเตอร์ VR แบบสามเฟส ซึ่งมีสเตเตอร์ 12 ฟัน และโรเตอร์

8 ฟัน ในรูปจะแสดงให้เห็นคอยล์ทุกอันที่ถูกกระตุ้นด้วยไฟฟ้าตามขั้นตอน และสำหรับคอยล์ที่ไม่ถูกกระตุ้น จะวาดให้ดูเพียงอันเดียวเพื่อความสะดวกต่อการทำความเข้าใจ ในรูปนี้แต่ละเฟสจะมีคอยล์ที่ใช้งาน 4 คอยล์ด้วยกัน ซึ่งสำหรับมอเตอร์สามเฟสก็จะมีคอยล์ทั้งหมด 12 อัน ตัวสเตเตอร์และโรเตอร์จะต้องสามารถทำให้เป็นแม่เหล็กได้เร็วเมื่อมีกระแสไฟไหลผ่านคอยล์ และทำให้หมดสภาพการเป็นแม่เหล็กได้เร็วเมื่อหยุดจ่ายไฟ ดังนั้นสเตเตอร์และโรเตอร์มักจะสร้างจากเหล็กซิลิกอนเพื่อให้มีคุณสมบัตินี้



รูป 2.8โครงสร้างภายในของเสต็ปปิ้งมอเตอร์แบบ VR

เมื่อเฟสใดเฟสหนึ่งของมอเตอร์ถูกกระตุ้นด้วยกระแสไฟตรง (Direct Current ,DC) ที่สเตเตอร์ก็จะเกิดความเป็นแม่เหล็กขึ้น และดึงฟันของโรเตอร์ที่อยู่ใกล้กับเฟสนั้นที่สุดให้มาตรงกับ

ฟันของสเตเตอร์ที่ถูกกระตุ้น เพราะตำแหน่งนี้คือตำแหน่งที่มี Reductance ต่ำที่สุดและมอเตอร์จะอยู่ในสภาพคงตัว (Equilibrium)

รูป 2.8 ก. แสดงให้เห็นตำแหน่งโรเตอร์เมื่อเฟส C ถูกกระตุ้น ถ้าหากว่าตัดไฟจากเฟส C และกระตุ้นเฟส B แทน ก็จะทำให้ฟันของโรเตอร์ซึ่งอยู่ใกล้เฟส B ที่สุดถูกดึงเข้ามาตรงกับฟันสเตเตอร์ของเฟส B ดังแสดงไว้ในรูป 2.8 ข. การเปลี่ยนเฟสนี้จะทำให้มอเตอร์หมุนไปหนึ่งขั้น ซึ่งในกรณีนี้ หนึ่งขั้นจะเท่ากับ 15^0 ในทิศทางทวนเข็มนาฬิกา

การกระตุ้นเฟส A ตามรูป 2.8 ค. จะทำให้มอเตอร์หมุนไปอีก 15^0 ในทิศทางทวนเข็มนาฬิกา แต่ถ้าหากว่ากระตุ้นเฟส C แทนที่จะกระตุ้นเฟส A มอเตอร์ก็จะหมุนกลับไปในทิศทางตามเข็มนาฬิกาหนึ่งขั้น และตำแหน่งของโรเตอร์จะกลับไปที่ตำแหน่งเดิมในรูป 2.8 ก.

เมื่อมีการกระตุ้นเฟส B และหยุดกระแสไฟที่เฟส C มอเตอร์จะหมุนไปหนึ่งขั้นเป็นมุมที่แน่นอน มุมนี้เรียกว่า มุมเสต็ป (Step Angle) และถ้าหากว่าเฟส A ถูกกระตุ้นและหยุดการกระตุ้นที่เฟส B ในเวลาต่อมา ก็จะทำให้มอเตอร์หมุนไปอีกหนึ่งเสต็ป ดังนั้นตำแหน่งเชิงมุมของโรเตอร์สามารถจะควบคุมได้ โดยการเลือกจ่ายกระแสไฟให้แต่ละเฟสตามลำดับ ถ้าหากว่าการสลับเฟสเป็นไปอย่างต่อเนื่อง ทิศทางการหมุนและความเร็วเฉลี่ยก็สามารถควบคุมได้โดยระบบสลับเฟส ซึ่งระบบสลับเฟสนี้อาจจะเป็นไมโครคอมพิวเตอร์หรือไมโครโปรเซสเซอร์ หรือระบบลอจิกไฟฟ้าก็ได้

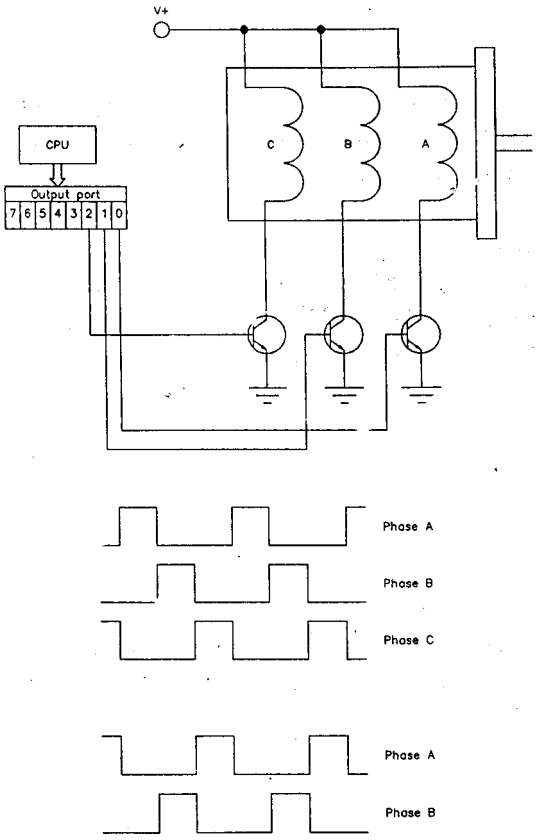
2.4.3 การขับมอเตอร์

ระบบมอเตอร์ได้แสดงไว้ในรูป 2.9 ระบบสร้างสัญญาณขับในรูปคือไมโครโปรเซสเซอร์ที่ส่งสัญญาณการขับออกมาที่พอร์ท และสัญญาณจากพอร์ทจะถูกขยายด้วยทรานซิสเตอร์ ซึ่งทำหน้าที่เป็นสวิทช์จ่ายไฟให้แก่เฟสต่าง ๆ ของมอเตอร์ ความถี่ของการสลับเฟสจะควบคุมความเร็วของมอเตอร์ จำนวนของการสลับเฟสจะควบคุมตำแหน่งของโรเตอร์และลำดับของการสลับเฟส (เฟสไหนก่อนเฟสไหนหลัง) จะควบคุมทิศทางการหมุน

วงจรในรูป 2.9 ยังใช้งานจริงไม่ได้ เพราะเมื่อทรานซิสเตอร์ตัดกระแสที่ผ่านคอยล์ที่มีความเหนี่ยวนำ จะทำให้เกิดแรงดันไฟฟ้ากลับทิศทางคร่อมคอยล์ และแรงดันนี้อาจสูงพอที่จะทำให้ทรานซิสเตอร์เสียหายได้ ในรูป 2.10 แสดงให้เห็นถึงโปรแกรมที่เขียนขึ้นด้วยภาษา Turbo Pascal ซึ่งใช้ในการควบคุมการหมุนของมอเตอร์แบบง่าย ๆ โปรแกรมนี้สมมติเอาว่าไมโครคอมพิวเตอร์เชื่อมต่อกับมอเตอร์ผ่านพอร์ทหมายเลข 310h และที่พอร์ทนี้สัญญาณที่ขับออกไปสามารถจะแช่ไว้ได้ (Latched) จนกว่าจะได้รับคำสั่งใหม่

ในวงจรตัวอย่างนี้ ทรานซิสเตอร์ที่ใช้เป็นแบบ NPN ซึ่งหมายความว่าสัญญาณที่พอร์ทจะต้องเป็นค่าสูง (1) เพื่อให้ทรานซิสเตอร์ปล่อยกระแสผ่าน ดังนั้นเพื่อให้มอเตอร์หมุนตาม

เข็มนาฬิกา สัญญาณเป็นเลขฐานสองที่จะออกไปที่พอร์ทควรจะเรียงลำดับ 001 ,010, และ 100 ซึ่งในตัวเลขฐานสิบก็คือ 1 ,2 และ 4 เมื่อต้องการให้มอเตอร์หมุนทวนเข็มนาฬิกา สัญญาณเป็นเลขฐานสองที่จะออกไปที่พอร์ทตามลำดับก็คือ 001 ,100 และ 010 เมื่อโปรแกรมนี้ทำงาน ก็จะสั่งงานให้พอร์ททำการจ่ายไฟไปที่เฟส A เพื่อหมุนโรเตอร์มาไว้ที่เฟส A หลังจากนั้นจึงไปรอรับข้อมูลว่าจะให้มอเตอร์หมุนกี่เสต็ป เมื่อผู้ใช้ใส่จำนวนเสต็ปแล้ว โปรแกรมก็จะเริ่มทำงานที่โพรซีเยอร์ Drivestep ภายในโพรซีเยอร์ นี้ จะเก็บข้อมูลไว้แล้วในอาเรย์ค่าคงที่ชื่อ Sphase ซึ่งแต่ละตัวของอาเรย์จะหมายถึงการขับเฟส A B และ C ถ้าหากว่าข้อมูลที่ส่งออกไปเป็นบวกโปรซิเตอร์นี้ก็จะสั่งให้มอเตอร์หมุนตามเข็มนาฬิกา และถ้าข้อมูลเป็นลบ มอเตอร์ก็จะหมุนทวนเข็มนาฬิกาไปจนครบจำนวนเสต็ปที่ใส่เข้าไป จากโปรแกรมนี้ ผู้อ่านจะสามารถเห็นได้ว่าความเร็วในการหมุนของมอเตอร์จะขึ้นอยู่กับค่าที่ใช้ในคำสั่ง Delay ทิศทางที่มอเตอร์หมุนจะขึ้นอยู่กับว่าข้อมูลที่ใส่เข้าไปเป็นบวกหรือลบ และตำแหน่งของโรเตอร์เมื่อโปรแกรมทำงานเสร็จแล้ว เทียบกับตำแหน่งของโรเตอร์ตอนเริ่มโปรแกรม จะขึ้นอยู่กับค่าของตัวเลขที่ใส่เข้าไป



รูป 2.9 ระบบขับเสต็ปปิ้งมอเตอร์

```
Program StepperTest1;
{This program is written with Turbo Pascal version 4.0}
uses crt;
var Phase,J : Integer;
        Procedure DriveStep(Steps : Integer);
            Const Sphase : Array[1..3] of byte = (1,2,4);
            Var N : Integer;
            Begin
              If Step = 0 then exit;

                  If Step > 0 then
                  For N := 1 to Steps do
                     Begin
                           Inc(Phase);
                           If  Phase > 3 then Phase := 1;
                          Port[$310] := Sphase[Phase];
                          Delay(10);
                       End
                    Else
                    For N := Steps to –1 do
                      Begin
                        Dec(Phase);
                        If Phase < 1 then Phase := 3 ;
                        Port[$310] := Sphase[Phase];
                        Delay(10);
                      End;
              End;
Begin
      Phase := 1;
      Port[$310] := 1;
      J := 0;
      While J <> -999 do
            Begin
                  Write (‘Enter number of steps or –999 to exit’);
                  ReadLn(J);
                  If  J <> -999 then DriveStep(J);
            End;
End
```

รูป 2.10 โปรแกรมควบคุมเสต็ปมอเตอร์อย่างง่าย

2.4.4 ตัวแปรที่สำคัญของมอเตอร์

ในส่วนนี้จะกล่าวถึงตัวแปรที่สำคัญของเสต็ปปิ้งมอเตอร์ ซึ่งตัวแปรบางตัวได้กล่าวถึงไปแล้วในตอนต้น

มุมเสต็ป (Step Angle) การขับเสต็ปปิ้งมอเตอร์นั้นกระทำได้โดยการกระตุ้นแต่ละเฟสตามลำดับและในการเปลี่ยนเฟสแต่ละครั้ง เพลาของมอเตอร์จะหมุนไปเป็นมุมที่มีค่าคงที่ มุมนี้เรียกว่ามุมเสต็ป หรือ θ_s เสต็ปมอเตอร์อาจมีมุมได้ดังต่อไปนี้ คือ 18^0 ,15^0,7.5^0 ,5^0,2.5^0,2.0^0,1.8^0, 0.9^0,0.72^0

จำนวนเสต็ปต่อรอบ ค่านี้คือ จำนวนเสต็ปรวมที่จะต้องใช้ในการหมุนมอเตอร์ไปหนึ่งรอบพอดี

อัตราการเสต็ป (Stepping Rate) ตัวเลขนี้คือ จำนวนเสต็ปต่อวินาที ซึ่งมอเตอร์ทำได้ที่แรงบิดใดแรงบิดหนึ่ง ถ้าหากว่าการขับมอเตอร์ใช้อัตราการขับสูงกว่าที่มอเตอร์จะทำได้ จะทำให้การควบคุมตำแหน่งหมุนของมอเตอร์พลาดและการทำงานไม่ได้ผล มอเตอร์พิเศษบางชนิดอาจมีอัตราการเสต็ปสูงถึง 20,000 เสต็ปต่อวินาทีก็ได้

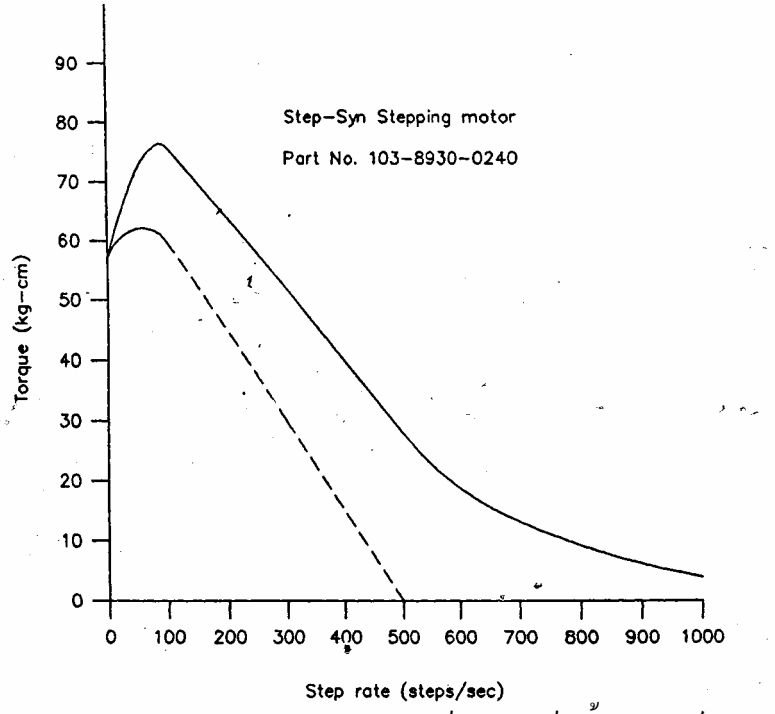
ความแน่นอนของเสต็ป (Step Accuracy) ความแน่นอนในการเสต็ปของมอเตอร์ทั่ว ๆ ไปจะอยู่ในช่วง $\pm$ 7 % ของมุมเสต็ป โดยความผิดพลาดนี้จะไม่ทวีคูณ ดังนั้นมอเตอร์ที่มีมุมเสต็ป 7.5^0 จะสามารถควบคุมตำแหน่งเชิงมุมได้โดยมีมีความแน่นอน 0.5^0 ไม่ว่าจะขับมอเตอร์ให้หมุนไป 7.5^0 หรือว่า $7,500^0$ หรือ 1000 เสต็ป ความผิดพลาดของมุมเสต็ปจะไม่ทวีคูณ และจะมีค่าเฉลี่ยเป็นศูนย์เมื่อแรงบิดคงที่ และจากการหมุนไปแล้วสี่เฟสสำหรับมอเตอร์แบบสี่เฟส นั้นเพราะว่าเมื่อหมุนไปสี่เสต็ป มอเตอร์จะกลับไปเฟสเดิมพอดี ซึ่งก็หมายความว่าโรเตอร์จะถูกดึงด้วยสเตเตอร์เดิมด้วยกระแสแรงดันเดิม ดังนั้นถ้าต้องการความแน่นอนสูง การขับควรจะกระทำเป็นค่าที่หารด้วยสี่ลงตัว

แรงบิด (Torque) แรงบิดที่มอเตอร์ใด ๆ สามารถทำได้จะขึ้นอยู่กับอัตราการเสต็ป กระแสที่ใช้ขับ วิธีการขับ และการออกแบบวงจร แรงบิดอาจแยกได้เป็นแรงบิดสถิตย์ (Static Torque หรือ Holding Torque) และแรงบิดจลน์ (Dynamic Torque)

แรงบิดสถิตย์ (Holding Torque) แรงบิดสถิตย์คือ แรงบิดที่กระทำจากภายนอกเพื่อให้มอเตอร์หมุนไปหนึ่งเสต็ปในขณะที่มอเตอร์หยุดนิ่ง และถูกกระตุ้นด้วยกระแสเต็มที่ตามข้อมูลจำเพาะของมอเตอร์ ดังนั้นแรงบิดสถิตย์อาจแปลได้ว่าเป็นความสามารถในการเบรค หรือยึดโรเตอร์ไว้ที่ตำแหน่งใดตำแหน่งหนึ่ง แรงบิดสถิตย์นี้มักจะมีค่าสูงกว่าแรงบิดขณะหมุน

แรงบิดจลน์ (Dynamic Torque) รูป 2.11 แสดงถึงกราฟแรงบิดกับอัตราการเสต็ปของเสต็ปปิ้งมอเตอร์ตัวหนึ่ง กราฟแรงบิดโดยทั่วไปแล้วจะมีสองเส้น เส้นหนึ่งคือ เส้นที่หมายถึง

ความสามารถของมอเตอร์ที่จะเริ่มหมุนโดยไม่มีการผิดพลาด (Start Without Error) ซึ่งบางครั้งเส้นนี้ก็เรียกว่าเป็นเส้นแรงบิดดึงเข้า (Pull-in Torque) กราฟเส้นนี้แสดงเป็นเส้นประในรูป อีกเส้นหนึ่งซึ่งเป็นเส้นทึบ คือแรงบิดขณะหมุน (Running Torque) ซึ่งบางครั้งเรียกว่าแรงบิดดึงออก (Pull-out Torque) แรงบิดนี้คือแรงบิดที่มอเตอร์ทำได้ขณะหมุน แต่ถ้ามอเตอร์เริ่มหมุนด้วยความเร็วนี้ทันทีอาจเกิดการหมุนข้ามเสต็ปได้ การที่จะไปถึงจุดแรงบิดขณะหมุนได้ การขับจะต้องค่อย ๆ เร่งมอเตอร์จากแรงบิดดึงเข้า จนกระทั่งความเร็วถึงจุดแรงบิดดึงออก กราฟความเร็วกับแรงบิดเป็นตัวแปรสำคัญในการเลือกมอเตอร์ให้เหมาะสมกับงานที่ทำ



รูป 2.11 กราฟแรงบิดกับความเร็วของเสต็ปปิ้งมอเตอร์

2.5 การเชื่อมต่อผ่านพอร์ทขนานของเครื่องคอมพิวเตอร์ส่วนบุคคล

2.5.1 ความรู้เบื้องต้นเกี่ยวกับพอร์ทขนาน

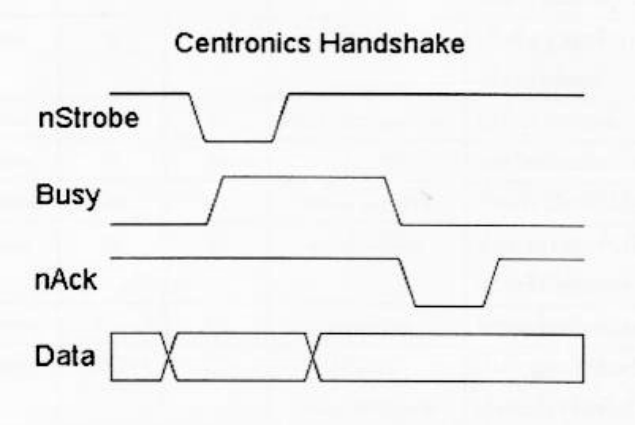
พอร์ทขนาน (Parallel Port) มีการถ่ายทอดข้อมูลเป็นแบบขนาน หรือเรียกอีกชื่อหนึ่งว่าพอร์ทเครื่องพิมพ์ (Line Printer Port : LPT) เนื่องจากพอร์ทนี้ใช้งานสำหรับต่อเครื่องพิมพ์นั่นเอง

ด้วยการถ่ายทอดข้อมูลแบบขนานนั่นเอง ทำให้มีอัตราการส่งถ่ายข้อมูลสูงกว่าการถ่ายข้อมูลแบบอนุกรมประมาณ 8-10 เท่า และการประมวลผลข้อมูลส่วนใหญ่จะมีขนาด 8 บิต ดังนั้นพอร์ทขนานจึงสามารถรองรับการถ่ายทอดข้อมูล 8 บิตได้โดยไม่ต้องเพิ่มเติมส่วนใด ๆ

2.5.2 ลักษณะทางกายภาพของพอร์ทขนาน

เพื่อให้เข้าใจในการนำพอร์ทขนานไปใช้งาน จึงต้องมาทำความเข้าใจก่อนว่า ปรกตินั้นการสั่งพิมพ์งานจากเครื่องคอมพิวเตอร์ไปยังพอร์ทขนานนั้นมีรูปแบบอย่างไร ในรูป 2.12 แสดงไดอะแกรมเวลาของการติดต่อระหว่างพอร์ทขนานกับเครื่องพิมพ์ ซึ่งจะเห็นได้ว่ามีสัญญาณที่ใช้งานจริง ๆ มีไม่มาก เริ่มจากสัญญาณพอร์ท Data ถูกส่งออกไปยังเครื่องพิมพ์ พร้อมทั้งส่งสัญญาณ Strobe ออกไปด้วย เพื่อให้เครื่องพิมพ์รับรู้ว่ามีการส่งสัญญาณใหม่มาที่ขา Data แล้ว จากนั้นคอมพิวเตอร์จะต้องรอการตอบกลับจากเครื่องพิมพ์ นั่นคือเครื่องพิมพ์จะสร้างสัญญาณ Busy เพื่อบอกว่าเครื่องพิมพ์ยังไม่พร้อมที่จะรับข้อมูลใหม่ จนกระทั่งเมื่อเครื่องพิมพ์พร้อม เครื่องพิมพ์จะสร้างสัญญาณ Ack ส่งไปยังเครื่องคอมพิวเตอร์เพื่อแจ้งว่า เครื่องพร้อมที่จะรับข้อมูลใหม่แล้ว

สัญญาณข้อมูลขนาด 8 บิต, สัญญาณ Stobe และสัญญาณ Ack (Acknowledge) เป็นสัญญาณที่สำคัญในการส่งข้อมูลจากคอมพิวเตอร์ไปยังเครื่องพิมพ์ นอกจากสัญญาณทั้ง 3 แล้ว ส่วนใหญ่การติดต่อกับเครื่องพิมพ์ยังมีสัญญาณอื่น ๆ ร่วมด้วย เนื่องจากเครื่องพิมพ์ทำหน้าที่ 3 อย่างด้วยกัน คือ รับข้อมูลจากเครื่องคอมพิวเตอร์, พิมพ์ข้อมูลที่รับเข้ามา และตอบสนองต่อการใช้งานของผู้ใช้ เช่น การเปลี่ยนรูปแบบตัวอักษร เป็นต้น ซึ่งสามารถสรุปหน้าที่ของสัญญาณที่จำเป็นต่อการติดต่อดังตาราง 2.1 และแสดงคุณสมบัติสัญญาณทั้งหมดที่ใช้งานโดยพอร์ทขนานไว้ดังตาราง 2.2



รูป 2.12 ไดอะแกรมเวลาของการส่งสัญญาณไปยังเครื่องพิมพ์

ตาราง 2.1 สัญญาณที่สำคัญของพอร์ทขนานที่ใช้ติดต่อกับเครื่องพิมพ์

สัญญาณ	หน้าที่การทำงาน	แหล่งกำเนิด
ข้อมูล 8 บิต	ข้อมูลที่ส่งจากคอมพิวเตอร์ไปยังเครื่องพิมพ์	คอมพิวเตอร์
Strobe	แจ้งเครื่องพิมพ์ถึงข้อมูลที่ส่งมาใหม่	คอมพิวเตอร์
Acknowledge	เครื่องพิมพ์แจ้งมายังคอมพิวเตอร์ว่าได้รับข้อมูลใหม่แล้ว	เครื่องพิมพ์
Busy	แจ้งสถานะว่าเครื่องพิมพ์ไม่ว่างจะรับข้อมูลใหม่	เครื่องพิมพ์
Error	แจ้งสถานะว่าเครื่องพิมพ์เกิดข้อผิดพลาด	เครื่องพิมพ์
Reset	รีเซตเครื่องพิมพ์	คอมพิวเตอร์

ตาราง 2.2 คุณสมบัติสัญญาณทั้งหมดที่ใช้งานโดยพอร์ทขนาน

Pin No.	Register	Direction	ตำแหน่งบิต	ชื่อขาสัญญาณ	หน้าที่การทำงาน
1	Control	In/Out	C0	nStrobe	แอกทีฟ “0” ส่งค่าออกไปเพื่อขา Data มีข้อมูลแล้ว
2-9	Data	Out	D1-D8	Data1-Data8	ส่งและรับเข้าข้อมูลเอาท์พุตและอินพุต
10	Status	In	S6	nAck	แอกทีฟ “0” ส่งสัญญาณมาจากเครื่องพิมพ์แสดงว่าได้รับข้อมูลที่ส่งไปแล้ว
11	Status	In	S7	Busy	สัญญาณแจ้งจากเครื่องพิมพ์ว่ายังไม่พร้อมรับข้อมูล
12	Status	In	S5	Paper-Out / Paper-End	แจ้งกระดาษหมด
13	Status	In	S4	Select	แจ้งว่าเครื่องพิมพ์ต่ออยู่
14	Control	In/Out	C1	nAuto-Linefeed	สั่งเครื่องพิมพ์ให้เลื่อนบรรทัด
15	Status	In	S3	nError/nFault	สัญญาณมาจกเครื่องพิมพ์ยทยังคอมพิวเตอร์เพื่อแสดงข้อผิดพลาดขากการพิมพ์

ตาราง 2.2 (ต่อ) คุณสมบัติสัญญาณทั้งหมดที่ใช้งานโดยพอร์ทขนาน

Pin No.	Register	Direction	ตำแหน่งบิต	ชื่อขาสัญญาณ	หน้าที่การทำงาน
16	Control	In/Out	C2	nInitialize	รีเซตเครื่องพิมพ์โดยใช้ลอจิก “0”
17	Control	In/Out	C3	nSelect-Printer/ nSelect-In	แจ้งการเลือกเครื่องพิมพ์ที่ต่ออยู่
18-25				Ground	กราวด์

2.5.3 การเขียนโปรแกรมติดต่อกับพอร์ทขนาน

พอร์ทขนานของเครื่องคอมพิวเตอร์จะมีลักษณะเช่นเดียวกันกันอุปกรณ์เอาท์พุตและอินพุตตัวอื่น ๆ คือ เมื่อต้องการติดต่อจะต้องกำหนดแอดเดรสที่ต้องการติดต่อด้วย ดังตาราง 2.3 แสดงแอดเดรสของพอร์ทขนาน

ตาราง 2.3 แสดงแอดเดรสของพอร์ทขนาน

แอดเดรส (Base Address)	การใช้งาน
3BCh-3BFh	ใช้เชื่อมต่อกับจอภาพ
378h-37Fh	แอดเดรสสำหรับ LPT1
278h-27Fh	แอดเดรสสำหรับ LPT2

หมายเหตุ อักษร h แสดงถึงเลขฐาน 16

เมื่อเริ่มใช้งานคอมพิวเตอร์ BIOS จะค้นหาหมายเลขของพอร์ทที่ใช้ติดต่อสื่อสารทั้งหมด รวมทั้งหมายเลขพอร์ทของ LPT1, LPT 2 และ LPT3 ในขั้นแรก BIOS จะหาพอร์ทที่แอดเดรส 3BCh ถ้าหากพบ BIOS จะกำหนดแอดเดรสสำหรับ LPT 1 หลังจากนั้นจะไปหาที่ตำแหน่ง 378h และกำหนดให้กับพอร์ทสื่อสารต่อ ๆ ไป ซึ่งอาจเป็น LPT1 หากไม่มีการติดตั้งการ์ดสำหรับจอภาพที่ตำแหน่ง 3BCh หลังจาก BIOS ได้กำหนดแอดเดรสเรียบร้อยแล้ว แอดเดรสทั้งหมดจะถูกเก็บไว้ในหน่วยความจำต่าง ๆ ดังตาราง 2.4

ตาราง 2.4 ตำแหน่งของหน่วยความจำที่เก็บค่าแอดเดรสของพอร์ทต่าง ๆ

Start Address	หน้าที่
0000:0408	Base Address ของ LPT1

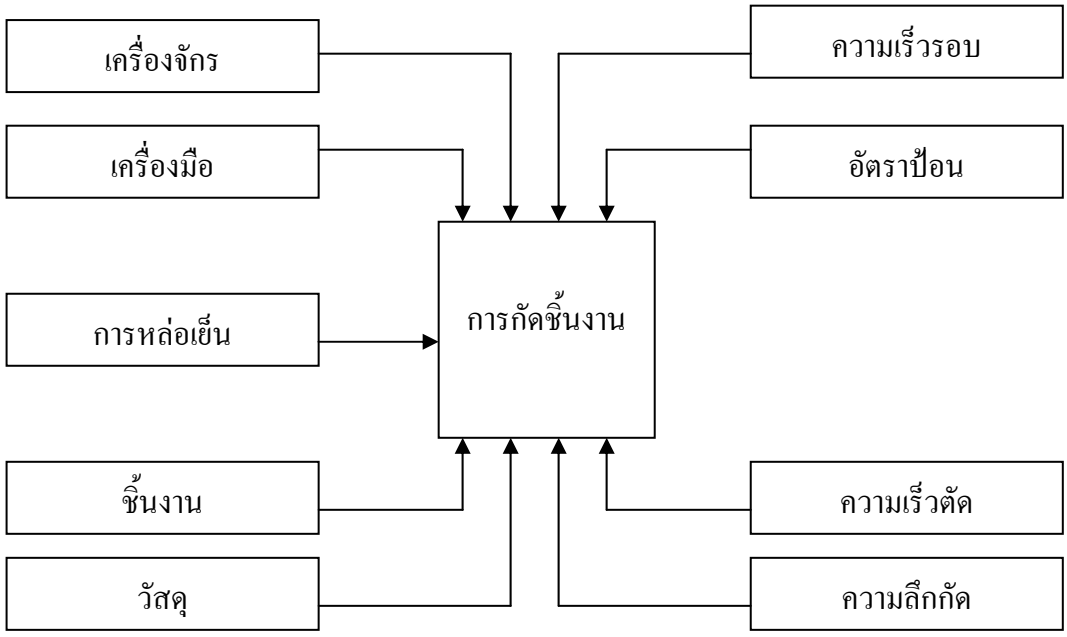
ตาราง 2.4 (ต่อ) ตำแหน่งของหน่วยความจำที่เก็บค่าแอดเดรสของพอร์ทต่าง ๆ

Start Address	หน้าที่
0000:040A	Base Address ของ LPT2
0000:040C	Base Address ของ LPT3
0000:040E	Base Address ของ LPT4

หมายเหตุ แอดเดรส 0000:040E ถูกใช้สำหรับพอร์ท PS2 และพอร์ทสื่อสารแบบใหม่

2.6 องค์ประกอบสำหรับงานกัด

ในการกัดชิ้นงานจำเป็นอย่างยิ่งที่เราจะต้องทราบข้อมูลต่าง ๆ ที่จำเป็นในการกัดชิ้นงาน ดังรูป 2.13 ให้ครบถ้วน โดยที่จะกล่าวในส่วนที่จำเป็นดังต่อไปนี้



รูป 2.13 องค์ประกอบต่าง ๆ ที่จำเป็นสำหรับงานกัด

2.6.1 ความเร็วตัด (V_s)

ความเร็วตัดของเครื่องกัด CNC จะขึ้นอยู่กับตัวแปรต่าง ๆ ดังนี้

ก. ชนิดของวัสดุงาน

วัสดุงานยิ่งมีความแข็งแรงสูง ก็ยิ่งใช้ค่าความเร็วกัดต่ำ โดยสังเกตได้จากตารางมาตรฐาน (อยู่ในภาคผนวก ก) การหาความเร็วรอบในการกัดชิ้นงาน หาได้จากสูตร

$$s = 1000\ V_s/\pi d \qquad \text{.....(8)}$$

โดยที่ d คือ ขนาดเส้นผ่านศูนย์กลางของมีดกัด (mm.)

s คือ ความเร็วรอบของมีดกัด (rev/min)

ข. ชนิดของมีดกัด

ชนิดของมีดกัดจะมีผลต่อความเร็วตัด โดยหากใช้มีดกัดที่ผลิตจากเหล็กรอบสูง จะใช้ความเร็วตัด (V_s) ประมาณ 12-14 m/min. และหากใช้มีดกัดที่ผลิตจากโลหะแข็ง จะใช้ความเร็วตัดประมาณ 70-240 m/min.

ค. อัตราการป้อนกัด (U)

อัตราการป้อนกัดสามารถดูได้ตากตารางมาตรฐาน (อยู่ในภาคผนวก ก) โดยอัตราป้อนของการกัดหยาบจะน้อยกว่าการกัดละเอียด

ง. ความลึกกัด

ความลึกของการกัดจะขึ้นอยู่กับการออกแบบชิ้นงาน ความแข็งแรง ความยาวของมีดกัด ลักษณะการทำงาน โดยการกัดหยาบจะตั้งค่าความลึกมากกว่าการกัดละเอียด

2.6.2 สัญลักษณ์ที่ใช้ในการเขียนคำสั่ง G code และ โปรแกรม NC

การเขียนคำสั่งเพื่อให้เครื่องกัด CNC ทำงานตามที่ผู้ใช้ต้องการ จำเป็นต้องมีรูปแบบการสั่งงานด้วยสัญลักษณ์และรูปแบบการสั่งงานด้วยรูปแบบมาตรฐาน แต่จะแตกต่างกันไปบ้างตามรูปแบบที่กำหนดโดยผู้ผลิตเครื่องจักร โดยที่สัญลักษณ์ต่าง ๆ มีความหมายที่แตกต่างกันไปดังแสดงในตาราง 2.5

ตาราง 2.5 สัญลักษณ์และความหมายที่ใช้ในคำสั่ง G code และโปรแกรม NC

สัญลักษณ์	**ความหมาย**
N	หมายเลขบรรทัด โดยปกติจะกำหนดให้เริ่มต้นบรรทัดที่ 5,10 หรือบรรทัดที่ 10,20,… เหตุผลที่ทิ้งช่วงบรรทัดเช่นนี้ เพื่อว่าในกรณีที่ต้องการเพิ่มเติมขั้นตอนการทำงาน จะได้แทรกโปรแกรมไว้ได้
G	คำสั่งการเคลื่อนที่ของหัวกัด
X,Y,Z	ระบบโคออดิเนต เป็นการกำหนดระยะทางหรือตำแหน่งที่หัวกัดจะต้องเคลื่อนไป
F (Feed)	อัตราป้อน โดยปกติอัตราป้อนในงานกัดจะมีหน่วยเป็น mm./min.

ตาราง 2.5 (ต่อ) สัญลักษณ์และความหมายที่ใช้ในคำสั่ง G code และโปรแกรม NC

สัญลักษณ์	**ความหมาย**
S (Speed)	ความเร็วรอบ มีหน่วยเป็น rev/min ในการป้อนโปรแกรม ความเร็วรอบ (N) จะแทนด้วย (S)
T (Tools)	เครื่องมือตัด เช่น มีดกัด ดอกสว่าน ดอกผายปาก ดอกเจาะนำศูนย์
M	คำสั่งเพิ่มเติม ทำได้หลายหน้าที่ เพื่อช่วยควบคุมหน้าที่ต่าง ๆ ร่วมกับคำสั่ง G