คำชี้แจง ให้ผู้เรียนทุกคนทำการทดลองตามใบงานการทดลองที่ 9 เรื่องงานโปรแกรมควมคุมสเต็ปเปอร์มอเตอร์ตามขั้นตอนการปฏิบัติงาน

จุดประสงค์ทั่วไป

เพื่อให้มีทักษะการปฏิบัติงานโปรแกรมควมคุมสเต็ปเปอร์มอเตอร์

จุดประสงค์การเรียนรู้เชิงพฤติกรรม (เพื่อให้ผู้เรียน.....)

1. สามารถใช้โปรแกรม Arduino IDE ในการเขียนโปรแกรมภาษา C เบื้องต้นได้อย่างถูกต้อง
2. สามารถใช้งานไมโครคอนโทรลเลอร์ บอร์ด Arduino UNO R3 เบื้องต้นได้อย่างถูกต้อง
3. สามารถประกอบและทดสอบวงจรควมคุมสเต็ปเปอร์มอเตอร์ได้อย่างถูกต้อง
4. สามารถเขียนโปรแกรมควมคุมสเต็ปเปอร์มอเตอร์ได้อย่างถูกต้อง
5. สามารถประยุกต์ใช้งานไมโครคอนโทรลเลอร์บอร์ด Arduino UNO R3 เบื้องต้นได้อย่างถูกต้อง
6. มีกิจนิสัยในการแสวงหาความรู้เพิ่มเติม การทำงานด้วยความประณีต รอบคอบและปลอดภัย

เครื่องมือและอุปกรณ์

1.	โปรแกรม Arduino IDE 1.8.4 หรือสูงกว่า	1	โปรแกรม
2.	สาย USB สำหรับ Arduino Uno R3	1	เส้น
3.	ชุดทดลอง Arduino Uno R3 พร้อมสายต่อวงจร	1	ชุด
4.	เครื่องคอมพิวเตอร์แบบพกพา	1	เครื่อง
5.	แผงต่อวงจร	1	ตัว
6.	มัลติมิเตอร์	1	ตัว
7.	เครื่องมือประจำตัว	1	ชุด

ข้อห้ามและข้อควรระวัง

1. ไม่เล่นและหยอกล้อกันในเวลาปฏิบัติงาน
2. ควรระวังไม่วางบอร์ด Arduino Uno R3 หรือชีลต่างๆ บนโต๊ะโลหะหรือที่วางที่เป็นโลหะ เพราะอาจเกิดการลัดวงจรของภาคจ่ายไฟได้
3. ไม่ควรต่อสายต่อวงจรในบอร์ด Arduino Uno R3 ทิ้งไว้ ควรถอดสายต่อวงจรออกให้หมด เพราะผล การทดลองอาจเกิดการผิดพลาดไม่เป็นไปตามทฤษฎีได้
4. ไม่ควรถอดสายสายโหลด USB เข้าออกตลอดเวลา เพราะอาจทำให้ภาคจ่ายไฟของบอร์ด Arduino Uno R3 เสียหายได้
5. ควรระวังเครื่องมือและอุปกรณ์เสียหายจากการปฏิบัติงานไม่ถูกต้องตามขั้นตอนและไม่ปลอดภัย

PATTAYA TECH

ทฤษฎี

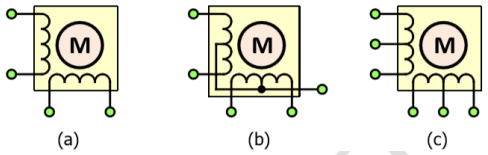
สเต็ปเปอร์มอเตอร์หรือที่บางคนเรียกว่าสเต็ปปิ้งมอเตอร์ เป็นมอเตอร์ที่ทำงานโดยอาศัยพลังงาน จากไฟฟ้ากระแสตรงเช่นเดียวกับมอเตอร์กระแสตรง แต่มีความความแตกต่างจากมอเตอร์กระแสตรงที่ เมื่อป้อนไฟให้กับสเต็ปเปอร์มอเตอร์จะขยับเพียงเล็กน้อยเพื่อเข้าสเต็ป ในขณะที่มอเตอร์ไฟฟ้า กระแสตรงทั่วไปจะหมุนตราบที่ยังมีพลังงานจ่ายให้แก่ตัวมัน ทั้งนี้เป็นเพราะโครงสร้างของสเต็ปเปอร์ มอเตอร์นั่นเอง โดยสเต็ปเปอร์มอเตอร์มีด้วยกัน 2 แบบคือ

1. Bipolar เป็นสเต็ปเปอร์มอเตอร์ที่มีขดลวด 2 ขดมีสายไฟให้ต่อใช้งาน 4 เส้นดังรูปที่ 9.11 (a)

2. Unipolar เป็นสเต็ปเปอร์มอเตอร์ที่มีขดลวด 4 ขด (2 ขดแบบมีแท็ปกลาง) โดยมี 2 แบบ

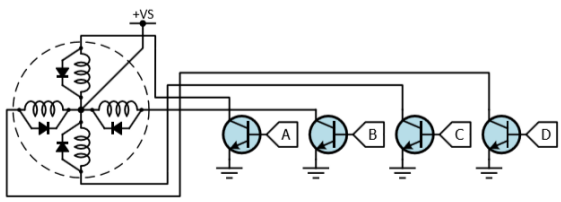
2.1 แบบมีสายไฟให้ต่อใช้งาน 5 เส้น ดังรูปที่ 9.1 (b)

2.2 แบบมีสายไฟให้ต่อใช้งาน 6 เส้น ดังรูปที่ 9.1 (c)



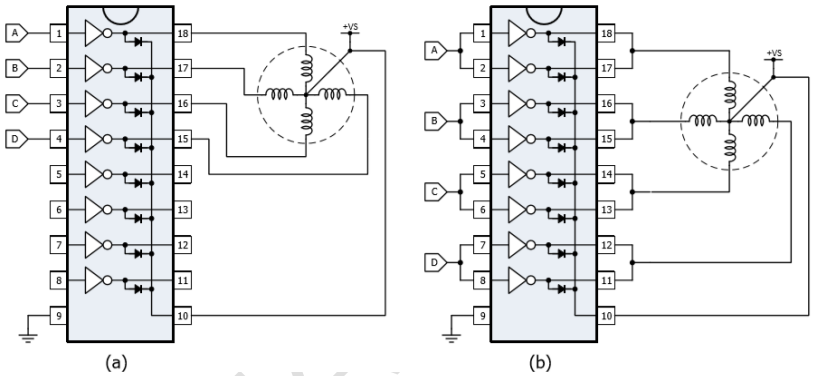
รูปที่ 9.1 แสดงโครงสร้างสเต็ปเปอร์ Bipolar และ Unipolar

ใบงานนี้เลือกใช้สเต็ปเปอร์มอเตอร์แบบ Unipolar ซึ่งมีขดลวดอยู่ภายในจำนวน 4 ขด (ที่จริง เป็น 2 ขดที่มีจุดแท็ปกลาง) โดยจำนวนรอบของขดลวดทั้ง 4 ขดจะมีค่าความต้านทานเท่ากัน ดังนั้นก่อนใช้งานต้องทำการหาตำแหน่งสายที่ใช้งานว่าเป็นสายตำแหน่งใดของขดลวด ค่าความต้านทานของ ขดลวดของสเต็ปเปอร์มอเตอร์ที่มีสายต่อแบบ 5 เส้นเมื่อวัดค่าความต้านทานของสายทุกเส้นจะสามารถ อ่านค่าได้ทุกเส้นนั่นหมายความว่าทุกเส้นเชื่อมต่อถึงกันและมีสายไฟเพียงเส้นเดียวเท่านั้นที่วัดค่าความ ต้านทานเทียบกับเส้นอื่น ๆ แล้วมีค่าความต้านทานที่วัดได้ในแต่ละเส้นเท่ากันทั้งหมด สายไฟเส้นนั้น เป็นสายร่วม นำสายร่วมต่อกับไฟบวก 5 โวลต์ นำสายไฟเส้นที่เหลือต่อลงกราวด์ครั้งละเส้นเรียงกัน โดย สลับกันไปมา จนกว่าสเต็ปเปอร์มอเตอร์หมุนไปทางด้านเดียวกัน สำหรับสเต็ปเปอร์มอเตอร์ ที่มีสายต่อ แบบ 6 เส้น เมื่อทำการวัดค่าความต้านทานจะมีเพียง 3 เส้น 2 ชุดที่วัดแล้วอ่านค่าความต้านทานได้ และ ในแต่ละชุดจะมีเส้นเดียวที่มีสายร่วม โดยเมื่อวัดค่าความต้านทานของสายร่วมกับสายเส้นอื่น จะมีค่า ความต้านทานเท่ากัน เมื่อนำมาใช้งานจะต้องนำสายร่วมทั้ง 2 เส้นของทั้ง 2 ชุดมาต่อร่วมกัน (แบบ 5 เส้นต่อไว้แล้วภายในตัวมอเตอร์) สำหรับวงจรขับต้องใช้วงจรขับที่ออกแบบมาสำหรับขับขดลวดเช่นเดียวกับวงจรขับรีเลย์โดยวงจรต้องมีไดโอดต่อคร่อมขดลวดเพื่อใช้ขจัดแรงดันย้อนกลับ (Back EMF) ที่เกิดขึ้นเมื่อทรานซิสเตอร์หยุดนำกระแส วงจรเป็นดังรูปที่ 9.2



รูปที่ 9.2 แสดงวงจรขบัสเต็ปเปอร์มอเตอร์ที่ใช้ทรานซิสเตอร์

ในทางปฏิบัติสามารถใช้ไอซีที่เป็นทรานซิสเตอร์อาร์เรย์อยู่ภายในแทนการใช้ทรานซิสเตอร์ได้ โดยไอซีนี้มีวงจรภายในเป็นวงจรทรานซิสเตอร์จำนวนหลายชุด ได้แก่เบอร์ ULN2003, ULN2803 ซึ่ง เบอร์ ULN2003 มีวงจรทรานซิสเตอร์ภายใน 7 ชุด และเบอร์ ULN2803 มี 8 ชุดโดยในแต่ละชุดเป็นวงจรทรานซิสเตอร์ที่มีการจัดวงจรภายในเป็นแบบวงจรทรานซิสเตอร์ที่ต่อแบบดาร์ลิงตัน ซึ่งสามารถขับ โหลดกระแสสูงโดยใช้กระแสเบสต่ำ (ลอจิกที่ป้อนเข้ามาทางอินพุท) จากลักษณะการทำงานของวงจรใน แต่ละชุดจะมีลักษณะการทำงานเช่นเดียวกับวงจรน็อตเกตในวงจรดิจิทัล สำหรับการขับสเต็ปเปอร์ มอเตอร์จะใช้งานเพียง 4 ชุดเท่านั้นดังรูปที่ 9.3 (a) ในกรณีที่ต้องการขับสเต็ปเปอร์มอเตอร์ที่มีกำลังมากเป็นมอเตอร์ที่ต้องการกระแสสูง ซึ่งการขับกระแสสูงสามารถใช้ไอซีเบอร์ ULN2803 มีวงจร ทรานซิสเตอร์ 8 ชุด โดยทำการขนานกัน 2 ชุดต่อการขับมอเตอร์ 1 ขด ดังรูปที่ 9.3 (b) ภายในไอซีมี ไดโอดป้องกันแรงดันย้อนกลับ (Back EMF) เตรียมไว้สำหรับโหลดที่เป็นขดลวด ดังนั้นเมื่อใช้งานจริง ไม่ต้องต่อไดโอดเพิ่มภายนอก เพียงแต่ต้องต่อขาร่วมของไดโอดเข้ากับแหล่งจ่ายที่เชื่อมเข้าที่จุดร่วมของ ขดลวด

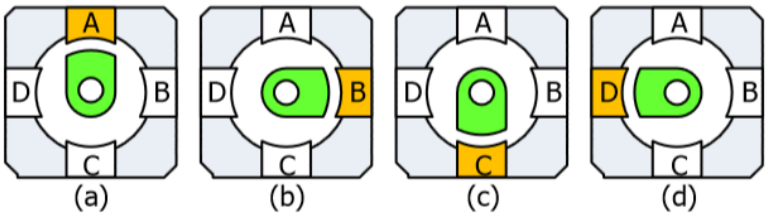


รูปที่ 9.3 แสดงวงจรขบัสเตป็เปอร์มอเตอร์แบบใช้ไอซี ULN2803

การขับสเต็ปเปอร์มอเตอร์

การขับสเต็ปเปอร์มอเตอร์เป็นการขับให้มอเตอร์ขยับไปตามจำนวนสเต็ปที่ต้องการซึ่งคุณสมบัติ ในการขยับในแต่ละสเต็ปของมอเตอร์มีหลายขนาดให้เลือกใช้งานเช่น 1.8 องศาต่อสเต็ป และ 7.5 องศา ต่อสเต็ป การควบคุมการหมุนของมอเตอร์สามารถควบคุมการทำงานได้ 3 แบบ คือ

1. **แบบเต็มสเต็ป 1 เฟส (Full step 1 phase)** เป็นการขับครั้งละ 1 เฟสเรียงกันไป ทำให้มอเตอร์ หมุนไปครั้งละ 1 สเต็ป ถ้ามอเตอร์มีคุณสมบัติ 1.8 องศาต่อสเต็ปก็จะหมุนไปครั้งละ 1.8 องศา โดยมีขั้นตอนขบัเฟสดังรูปที่ 9.4

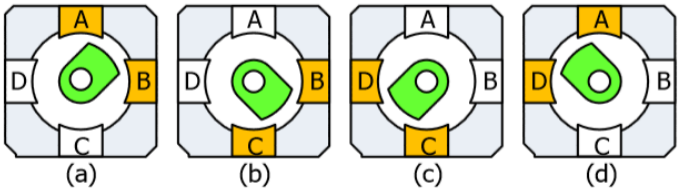


รูปที่ 9.4 แสดงการเคลื่อนที่เมื่อขับมอเตอร์แบบเต็มสเต็ป 1 เฟส

ตารางที่ 9.1 รูปแบบการกระตุ้นขดลวดของมอเตอร์แบบเต็มสเต็ป 1 เฟส

สเต็ปที่	ขดลวด A	ขดลวด B	ขดลวด C	ขดลวด D	รูปที่ 9.4
1	ON	OFF	OFF	OFF	(a)
2	OFF	ON	OFF	OFF	(b)
3	OFF	OFF	ON	OFF	(c)
4	OFF	OFF	OFF	ON	(d)

2. แบบเต็มสเต็ป 2 เฟส (Full step 2 phase) เป็นการขับครั้งละ 2 เฟส โดยมีคุณสมบัติที่ดีขึ้นคือ แรงบิดมากขึ้นเนื่องจากมีการขับครั้งละ 2 เฟส ทำให้เกิดสนามแม่เหล็กมากขึ้นนั่นเอง แต่มุมในการหมุน จะมีค่าเท่ากันกับแบบเต็มสเต็ป 1 เฟส โดยมีขั้นตอนขับเฟสดังรูปที่ 9.5

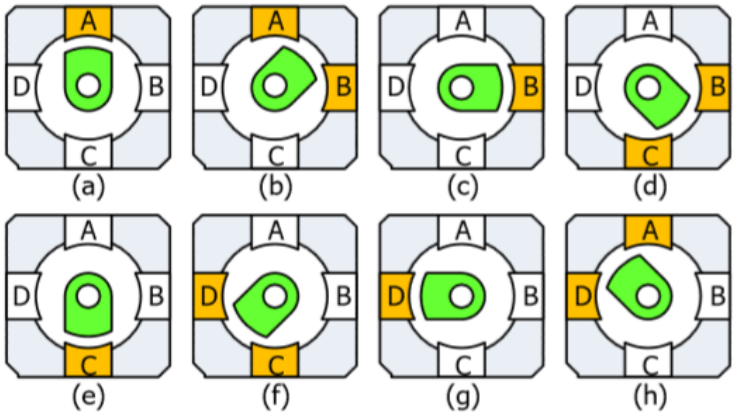


รูปที่ 9.5 แสดงการเคลื่อนที่เมื่อขับมอเตอร์แบบเต็มสเต็ป 2 เฟส

ตารางที่ 9.2 แสดงรูปแบบการกระตุ้นขดลวดของมอเตอร์แบบเต็มสเต็ป 2 เฟส

สเต็ปที่	ขดลวด A	ขดลวด B	ขดลวด C	ขดลวด D	รูปที่ 9.5
1	ON	ON	OFF	OFF	(a)
2	OFF	ON	ON	OFF	(b)
3	OFF	OFF	ON	ON	(c)
4	ON	OFF	OFF	ON	(d)

3. แบบครึ่งสเต็ป (Haft step) เป็นการควบคุมให้มอเตอร์หมุนไปครั้งละครึ่งของสเต็ปดังนั้นหาก มอเตอร์มอเตอร์มีคุณสมบัติ 1.8 องศาต่อสเต็ปก็จะหมุนไปครั้งละ 0.9 องศาทำได้ตำแหน่งที่เที่ยงตรงมาก ขึ้นเมื่อนำไปประยุกต์ใช้งานวิธีการขับจะใช้การผสมกันระหว่างแบบเต็มสเต็ป 1 เฟสกับแบบเต็มสเต็ป 2 เฟส ซึ่งมีขั้นตอนขับเฟสดังรูปที่ 9.6



รูปที่ 9.6 แสดงการเคลื่อนที่เมื่อขับมอเตอร์แบบครึ่งสเต็ป

ตารางที่ 9.3 แสดงรูปแบบการกระตุ้นขดลวดของมอเตอร์แบบครึ่งสเต็ป

สเต็ปที่	ขดลวด A	ขดลวด B	ขดลวด C	ขดลวด D	รูปที่ 9.6
1	ON	OFF	OFF	OFF	(a)
2	ON	ON	OFF	OFF	(b)
3	OFF	ON	OFF	OFF	(c)
4	OFF	ON	ON	OFF	(d)
5	OFF	OFF	ON	OFF	(e)
6	OFF	OFF	ON	ON	(f)
7	OFF	OFF	OFF	ON	(g)
8	ON	OFF	OFF	ON	(h)

ฟังก์ชั่น Arduino ที่ใช้งานในใบงานการทดลอง

1. **ฟังก์ชั่นกำหนดโหมดการทำงานให้กับขาพอร์ต** โดยสามารถกำหนดได้ทั้งขาดิจิทัลโดยใส่เพียงตัวเลขของขา (0, 1, 2,...13) และขาแอนาลอกที่ต้องการให้ทำงานในโหมดดิจิทัลแต่ การใส่ขาต้องใส่ A นำ หน้าซึ่งใช้ได้เฉพาะ A0, A1,...A5 ส่วนขา A6 และ A7 ไม่สามารถใช้งานในโหมดดิจิทัลได้ รูปแบบของฟังก์ชั่นเป็นดังนี้

   pinMode(pin,mode);

   pin : หมายเลขขาที่ต้องการเซตโหมด,mode : INPUT, OUTPUT, INPUT_PULLUP

2. **ฟังก์ชั่นส่งค่าลอจิกดิจิทัลไปยังขาพอร์ต** ค่า HIGH เป็นการส่งลอจิก 1 และค่า LOW เป็นการ ส่งลอจิก 0 ออกไปยังขาพอร์ต ฟังก์ชั่นนี้จะทำงานได้ต้องมีการใช้ฟังก์ชั่น pinMode ก่อน รูปแบบของฟังก์ชั่นเป็นดังนี้

   digitalWrite(pin,value);

   pin : หมายเลขขาที่ต้องการเขียนลอจิกออกพอร์ต ,value : HIGH หรือ LOW

3. **ฟังก์ชั่นหน่วงเวลาหรือฟังก์ชั่นหยุดค้าง** การใช้งานสามารถกำหนดตัวเลขของเวลาที่ต้องการหยุดค้าง ตัวเลขที่ใส่เป็นตัวเลขของเวลาหน่วยเป็นมิลลิวินาที ตัวเลขของเวลาที่ใส่ ได้สูงสุดคือ 4,294,967,295 ซึ่งเป็นขนาดของตัวแปร unsigned long รูปแบบของฟังก์ชั่นเป็นดังนี้

   Delay(ms); ms : ตัวเลขที่หยุดค้างของเวลาหน่วยมิลลิวินาที (unsigned long)

4. **ฟังก์ชั่นกำหนดความเร็วในการสื่อสารทางพอร์ตอนุกรม** รูปแบบของฟังก์ชั่นเป็นดังนี้

   Serial.begin(speed); speed: ตัวเลขของอัตราเร็วในการสื่อสารผ่านพอร์ตอนุกรม

5. **ฟังก์ชั่นส่งข้อมูลออกพอร์ต** เป็นฟังก์ชั่นที่ใช้ในการส่งข้อมูลออกทางพอร์ตอนุกรมหรือพิมพ์ข้อมูลออกทางพอร์ตเพื่อแสดงผลที่จอคอมพิวเตอร์เมื่อพิมพ์เสร็จตัวเคอร์เซอร์จะรออยู่ที่ท้ายสิ่งที่พิมพ์นั้น ๆ รูปแบบของฟังก์ชั่นเป็นดังนี้

   Serial.print(val); Serial.print(val, format);

PATTAYA TECH

6. ฟังก์ชั่นส่งข้อมูลออกพอร์ต คล้ายกับฟังก์ชั่น Serial.print ต่างกันตรงที่เมื่อพิมพ์เสร็จตัวเคอร์เซอร์จะขึ้นมารอยังบรรทัดใหม่ ดังนั้นเมื่อสั่งพิมพ์ครั้งถัดไปข้อมูลที่ปรากฏจะอยู่ที่บรรทัดใหม่แทนที่จะต่อท้ายเหมือนกับฟังก์ชั่น Serial.print รูปแบบของฟังก์ชั่นเป็นดังนี้

Serial.println(val); Serial.println(val, format);

7. ฟังก์ชั่นอ่านสัญญาณแอนาลอก เป็นฟังก์ชั่นที่อ่านสัญญาณแอนาลอกที่ปรากฏอยู่ที่ขาพอร์ต แอนาลอกที่ต้องการอ่านนั้น ๆ ค่าที่อ่านได้จะอยู่ในช่วง 0-1023 สำหรับแรงดันของสัญญาณแอนาลอกที่ 0-5V ดังนั้น ต้องใช้ตัวแปรที่เป็น int สำหรับเก็บค่าที่อ่านได้ รูปแบบของฟังก์ชั่นเป็นดังนี้

analogRead(pin); pin: ขาพอร์ตแอนาลอกที่ต้องการอ่านค่าสัญญาณแอนาลอก

8. ฟังก์ชั่นแปลงช่วงตัวเลข เป็นฟังก์ชั่นทำหน้าที่เปลี่ยนแปลงค่าที่ได้รับจากตัวแปรจากช่วงตัวเลขระหว่างค่าหนึ่งถึงอีกค่าหนึ่งไปสู่ช่วงตัวเลขใหม่ที่ต้องการ

map(value, fromLow, fromHigh, toLow, toHigh)

value : ตัวแปรที่ต้องการอ่านค่านำมาเปลี่ยนช่วงตัวเลข

fromLow : ตัวเลขสเกลล่างสุดของค่าจากตัวแปร

fromHigh : ตัวเลขสเกลสูงสุดของค่าจากตัวแปร

toLow : ตัวเลขสเกลล่างสุดของค่าที่ต้องการเปลี่ยนไป

toHigh : ตัวเลขสเกลสูงสุดของค่าที่ต้องการเปลี่ยนไป

[ที่มา:ครูประภาส สุวรรณเพชร,**เอกสารประกอบการอบรม เรียนรู้และลองเล่น Arduino เบื้องต้น (ฉบับปรับปรุงครั้งที่ 1)** ,หน้าที่ 184-188.]

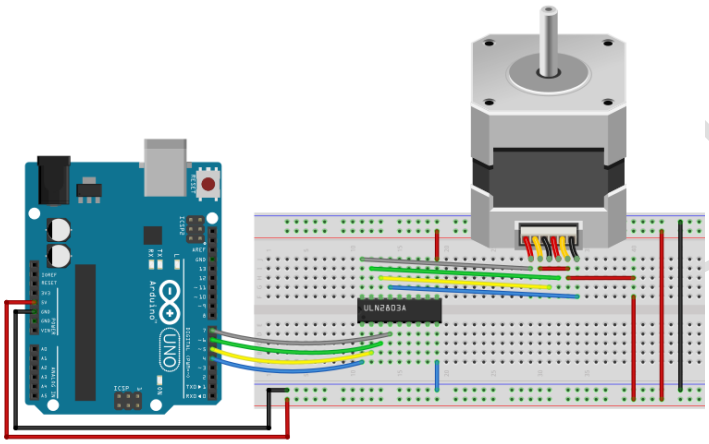
PATTAYA TECH

ลำดับขั้นการทดลอง

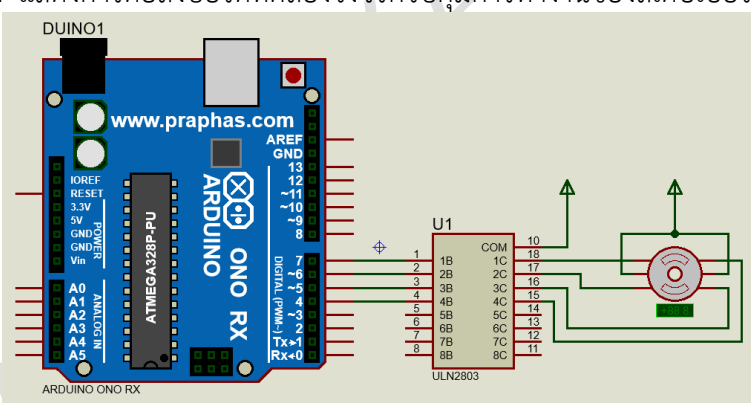
ตอนที่ 1 โปรแกรมควบคุมการทำงานของสเต็ปเปอร์มอเตอร์

แนวคิดการเรียนรู้ คือ เขียนโปรแกรมควบคุมการทำงานของสเต็ปเปอร์มอเตอร์โดยให้หมุนในทิศทางเดียวเป็น จำนวน 360 โดยมีขั้นตอนดังนี้

1. ประกอบวงจรควบคุมการทำงานของสเต็ปเปอร์มอเตอร์โดยใช้บอรด Arduino UNO R3 ดังรูปที่ 9.7

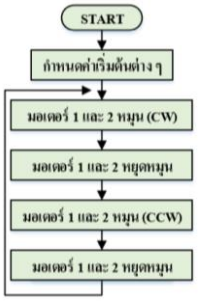


รูปที่ 9.7 แสดงการต่อลงบอร์ดทดลองวงจรควบคุมการทำงานของสเต็ปเปอร์มอเตอร์



รูปที่ 9.8 แสดงการต่อวงจรทดลองในโปรแกรมจำลองการทำงานของสเต็ปเปอร์มอเตอร์

2. เปิดโปรแกรม Arduino IDE จากนั้นพิมพ์โค้ดโปรแกรมการทำงานของสเต็ปเปอร์มอเตอร์ ตามรูปที่ 9.9 ดังต่อไปนี้



(ก) ผังงาน

Lab9-1

```
#define phaseA 4
#define phaseB 5
#define phaseC 6
#define phaseD 7
char PinCon[] = {phaseA,phaseB,phaseC,phaseD};
byte pattern[]={
                0B1000,
                0B0100,
                0B0010,
                0B0001};
void send2port(byte data);
void setup()
{
  for(char i=0;i<4;i++)
  {
    pinMode(PinCon[i],OUTPUT);
  }
}
void loop()
{
  for(int j=0;j<50;j++)
  {
     for(char i=0;i<4;i++)
     {
       send2port(pattern[i]);
       delay(55);
     }
   }
   send2port(0B0000);
   while(1);
}
void send2port(byte data)
{
  digitalWrite(PinCon[0],(data & 1 )? HIGH:LOW);
  digitalWrite(PinCon[1],(data & 2 )? HIGH:LOW);
  digitalWrite(PinCon[2],(data & 4 )? HIGH:LOW);
  digitalWrite(PinCon[3],(data & 8 )? HIGH:LOW);
}
```

(ข) โค้ดโปรแกรม

รูปที่ 9.9 แสดงโปรแกรมการทำงานของสเต็ปเปอร์มอเตอร์

3. บันทึกไฟล์โค้ด ชื่อ Lab9-1
4. ทำการ Compile โค้ด Lab9-1
5. เชื่อมต่อสาย USB กับ บอร์ด Arduino Uno R3

6. Upload โปรแกรม Lab9-1 ลงบอรด Arduino UNO R3
7. สังเกตวงจรการทำงานและบันทึกผลการทดลอง

.....

.....

.....

.....

.....

.....

.....

.....

8. คำถามท้ายการทดลองตอนที่ 1 จากโค้ดโปรแกรม Lab9-1 จงตอบคำถามต่อไปนี้
   8.1. บรรทัดที่ 1-4 ทำหน้าที่.....
   8.2. บรรทัดที่ 6-10 ทำหน้าที่.....
   8.3. บรรทัดที่ 14-17 ทำหน้าที่.....
   8.4. บรรทัดที่ 21-28 ทำหน้าที่.....
   8.5. บรรทัดที่ 33-38 ทำหน้าที่.....

ตอนที่ 2 งานที่มอบหมาย

เขียนโปรแกรมควบคุมการหมุนของสเต็ปเปอร์มอเตอร์ โดยให้มีการหมุนทวนเข็มและตามเข็มนาฬิกาในองศาที่กำหนดขึ้นเอง วงจรที่ใช้ทดลองให้ใช้วงจรที่ 9.7 โดยมีเงื่อนไขดังนี้ดังนี้

9. จงเขียนผังงานจากงานที่มอบหมาย

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

10. พิมพ์โค้ดโปรแกรมตามผังงานในข้อที่ 9
11. บันทึกไฟล์โค้ด ชื่อ Lab9-2

PATTAYA TECH

12. ทำการ Compile โค้ด Lab9-2
13. เชื่อมต่อสาย USB กับ บอร์ด Arduino Uno R3
14. Upload โปรแกรม Lab9-2 ลงบอรด Arduino UNO R3
15. สังเกตวงจรการทำงานและบันทึกผลการทดลอง

.....
.....
.....
.....
.....
.....
.....
.....

16. สรุปผลการทดลอง

.....
.....
.....
.....
.....
.....
.....
.....
.....
.....

PATTAYATECH