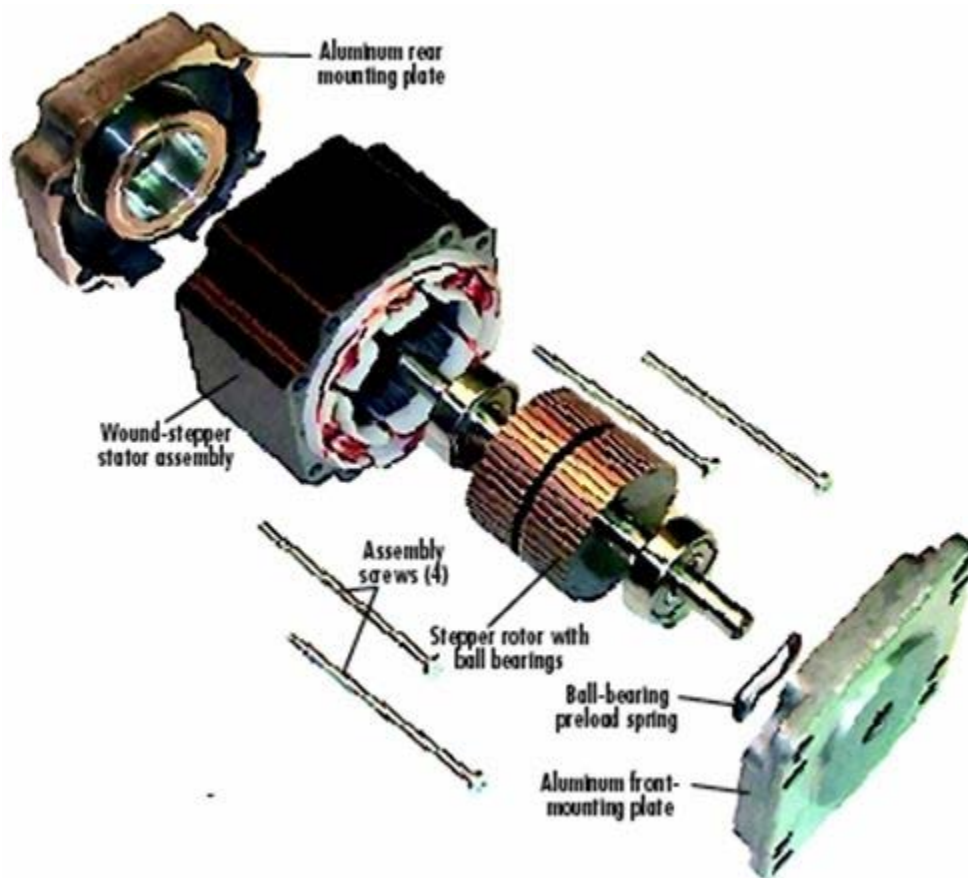
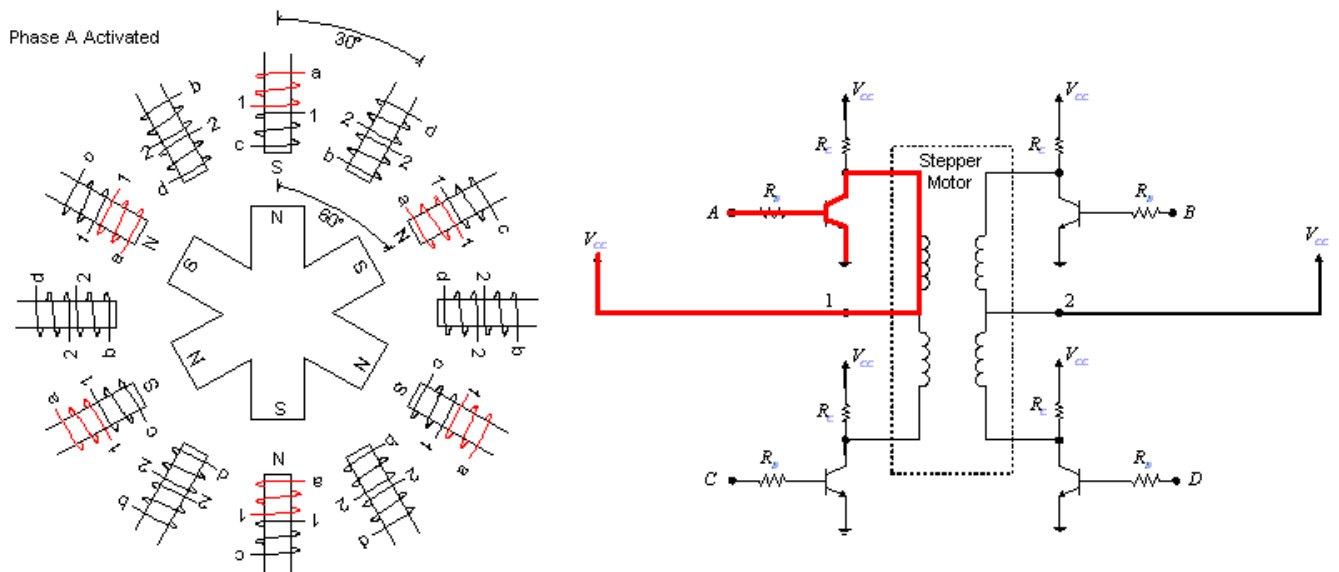


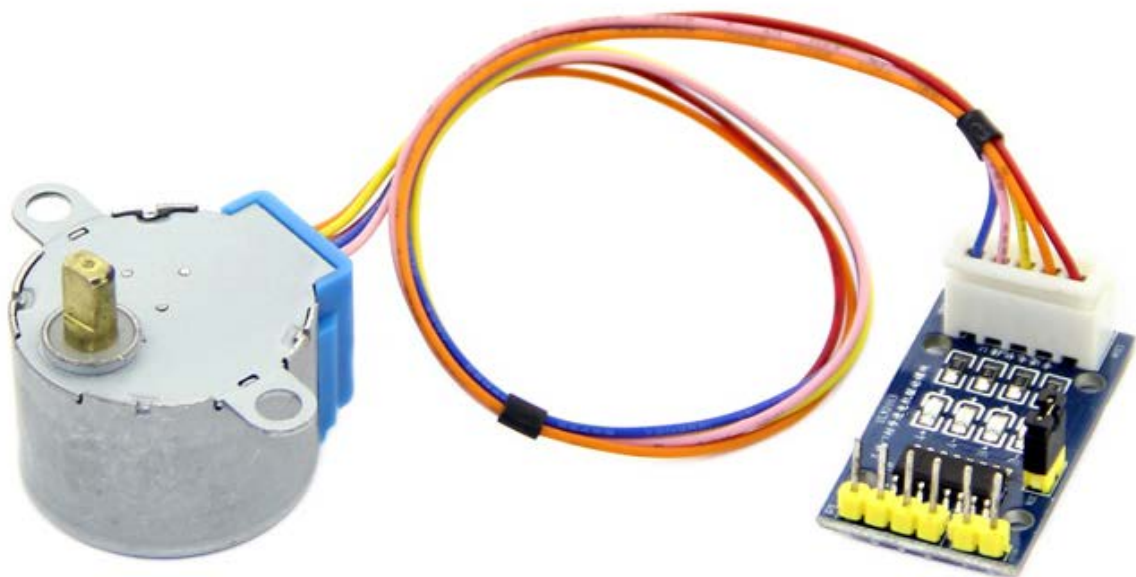
Stepper Motor หรือ **Stepping Motor** คือ มอเตอร์ที่มีการหมุนเป็นขั้นๆ (Step) เมื่อมีสัญญาณไฟฟ้าที่เป็นพัลส์มาป้อนเข้าที่สเต็ปเปอร์มอเตอร์จะทำให้แกนกลางของมอเตอร์หมุนเป็นมุมคงที่ คือมีค่าเป็นองศาต่อสเต็ป ทำให้สามารถควบคุมตำแหน่งของการหมุนได้อย่างแม่นยำ โดยไม่ต้องใช้การควบคุมแบบป้อนกลับ (Feedback Control) หรืออาศัยตัวตรวจจับการหมุนมาควบคุมตำแหน่ง การควบคุมสเต็ปเปอร์มอเตอร์จะใช้ไมโครคอนโทรลเลอร์ส่งสัญญาณดิจิทัลมาควบคุมบังคับทิศทาง และความเร็วในการหมุนของแกนสเต็ปเปอร์มอเตอร์ได้โดยตรง โดยไม่ต้องอาศัยแปรงถ่านจึงไม่เกิดการสึกหรอและไม่มีสัญญาณรบกวนที่เกิดจากหน้าสัมผัสของแปรงถ่าน จึงเป็นที่นิยมใช้ในอุปกรณ์ที่ต้องการควบคุมตำแหน่งและมุมได้อย่างแม่นยำ เช่น ปริ้นเตอร์ สแกนเนอร์ ฮาร์ดดิสก์ กล้องวงจรปิด เครื่องปรับอากาศ และอุปกรณ์ของรถยนต์ เป็นต้น โดยที่ Stepping Motor แต่ละตัวจะมีความละเอียดของมุมหมุนที่แตกต่างกันขึ้นอยู่กับโครงสร้างการผลิต



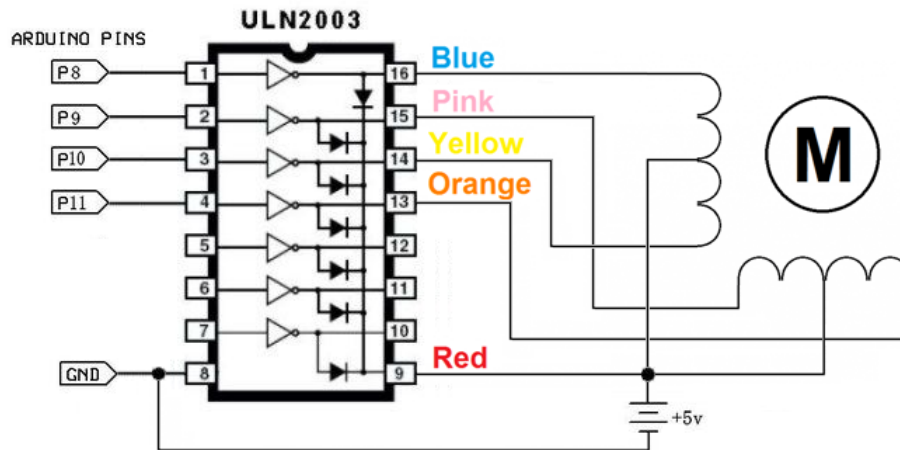
การทำงานของ Stepper Motor จะต้องป้อนแรงดันไฟฟ้าไปที่ขดลวดที่ติดตั้งบนสเตเตอร์ ให้ถูกต้องตามจังหวะเพื่อไปบังคับให้แม่เหล็กถาวร (Permanent Magnet) บนแกนโรเตอร์หมุนไปตามทิศทางที่กำหนด โดยทั่วไปสเต็ปเปอร์มอเตอร์แบ่งได้ 2 แบบ ตามลักษณะของโครงสร้างการต่อขดลวดภายในมอเตอร์ คือ Unipolar และ Bipolar ซึ่งหลักในการขับของ Stepper Motor ทั้งสองแบบทำงานจะทำงานคล้ายกัน คือการป้อนพัลส์เป็นช่วงๆเข้าไปยังขดลวดต่างๆ เพื่อให้ Stepper Motor หมุนไปตามองศาที่ต้องการ โดยปกติแบบ Bipolar จะมีสายไฟต่อ 4 เส้น การต่อวงจรอาจจะต้องใช้วงจร H-Bridge เข้ามาช่วยเพื่อกลับทิศทางของสนามแม่เหล็กภายใน ส่วนแบบ Unipolar จะมีสายไฟต่อ 5 เส้น แต่การทำวงจรควบคุมจะทำได้ง่ายกว่า เนื่องจากไม่ต้องกลับทิศของกระแสไฟที่ป้อนเข้าไปที่ขดลวด



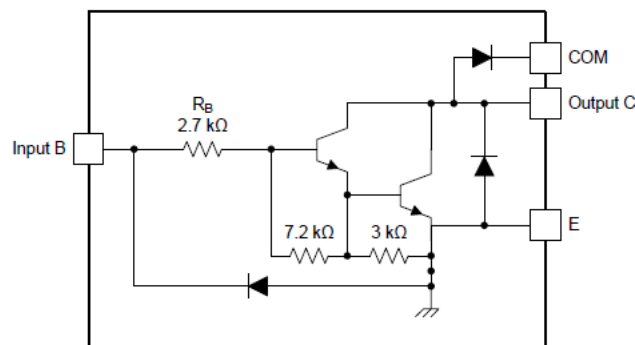
การทำงานของ Stepper Motor นั้นต้องใช้กระแสสูงเพื่อป้อนเข้าขดลวดให้ทำงาน ดังนั้น Microcontroller จะไม่สามารถเชื่อมต่อโดยตรงได้ จึงต้องมีชุดขับกระแสใช้ไอซีเบอร์ **ULN2003** การทดลองจะใช้วงจรควบคุมที่เป็นโมดูล 4 Phase Stepper Motor Driver ตามในรูป



การทำงานของวงจรจะจ่ายแรงดันไฟฟ้า 5V เข้าที่จุดต่อร่วม Common Anode ไปยัง Stepping Motor และป้อนสัญญาณจาก Microcontroller เป็นลอจิกต่าง ๆ จำนวน 4 บิต เข้าไปยัง IC 2003 ที่ทำหน้าที่เป็น Driver เพื่อขับเฟสให้กับ Stepping Motor ทั้ง 4 เส้นเพื่อให้ลงกราดตามจังหวะของสัญญาณที่ป้อน ซึ่งสามารถดูการทำงานทั้ง 4 เฟสได้จาก LED ที่ต่ออยู่บนบอร์ด โดยมีวงจรดังรูป



โดยที่วงจรภายในแต่ละขาจะใช้ทรานซิสเตอร์ต่อเป็นวงจรดาร์ลิ่งตัน (Darlington) เพื่อขับกระแสทำให้สามารถใช้กระแสได้ถึง 500 mA และมีชื่อทักไดโอด (Schottky Diode) ซึ่งเป็นไดโอดที่มีค่าแรงดันตกคร่อมขณะนำกระแสต่ำและทำงานได้ดีที่ความถี่สูง มาทำหน้าที่ป้องกันแรงดันไฟย้อนกลับ (Negative Undershoot) ที่เกิดจากการทำงานของมอเตอร์ ซึ่งจะเป็นอันตรายทำให้วงจรควบคุมเสียหายได้



ULN2003A Block Diagram

- ✓ 1. ให้เชื่อมต่อ Steper Motor กับวงจร 4 Phase Stepper Motor Driver Module เข้ากับบอร์ด Arduino ทางขา 8 ถึง 11 และต่อแหล่งจ่ายไฟฟ้า
- ✓ 2. ป้อนโปรแกรมเพื่อกำหนดการทำงานของ Stepping Motor ดังนี้

```
//declare variables for the motor pins
```

```
int motorPin1 = 8;
```

```
int motorPin2 = 9;
```

```
int motorPin3 = 10;
```

```
int motorPin4 = 11;
```

```
// Blue
```

```
// Pink
```

```
// Yellow
```

```
// Orange
```

```
// Red
```

```
int motorSpeed = 100;
```

```
int stepCount = 0;
```

```
//variable to set stepper speed
```

```
// number of steps the motor has taken
```

```
void setup()
```

```
{
```

```
pinMode(motorPin1, OUTPUT);
```

```
pinMode(motorPin2, OUTPUT);
```

```
pinMode(motorPin3, OUTPUT);
```

```
pinMode(motorPin4, OUTPUT);
```

```
Serial.begin(115200);
```

```
//declare the motor pins as outputs
```

```
// initialize the serial port:
```

```
}
```

```

void loop()
{
    wavedrive();
    fullstep();
    halfstep();

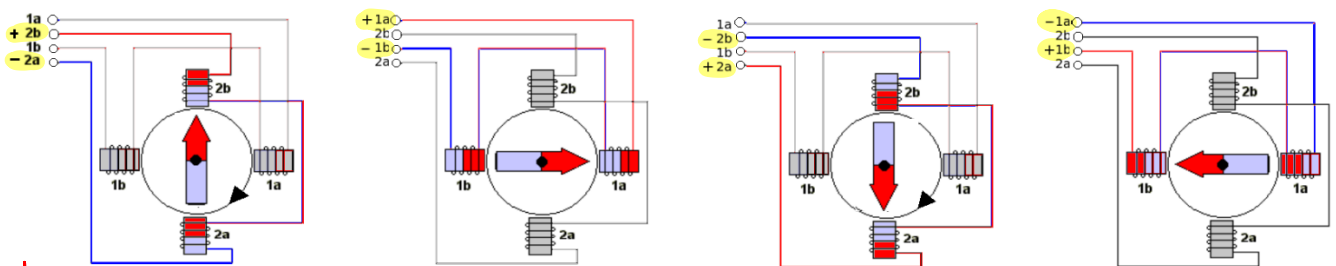
    Serial.print("steps: ");
    Serial.println(stepCount);
    stepCount++;
}

```

// จ่ายไฟให้ทำงานครึ่งละ 1 ขด
// จ่ายไฟให้ทำงานครึ่งละ 2 ขด
// ใช้ทั้งสองแบบรวมกันทำให้หมุนได้ครึ่งละครึ่งเสดป

การควบคุมการหมุนของสเต็ปเปอร์มอเตอร์ สามารถทำได้โดยการป้อนแรงดันไฟฟ้าคงที่เข้าไปที่ขั้วของขดลวดที่ควบคุมการหมุน เพื่อบังคับให้แม่เหล็กถาวรบนแกนโรเตอร์หมุนไปตามทิศการบังคับของขดลวดที่ติดตั้งบนสเตเตอร์ โดยจะต้องป้อนแรงดันให้ถูกต้องตามจังหวะเพื่อให้แกนโรเตอร์หมุนดังรูป

ตามตัวอย่างขดลวดแต่ละขดห่างกัน 90 องศา การหมุนแบบง่ายที่สุดทำได้โดยการจ่ายกระแสไฟเข้าไปกระตุ้นที่ขดลวดในแต่ละเฟสตามลำดับ 1a 2a 1b 2b ถ้าหากต้องการให้กระแสไหลในเฟสใดก็จะทำให้สถานะของเฟสนั้นเป็น High ซึ่งจะทำให้เกิดสนามแม่เหล็ก เพื่อไปดูดแม่เหล็กถาวรที่อยู่บนโรเตอร์ให้เคลื่อนที่ โดยมีทิศทางการหมุนตามลำดับการจ่ายกระแสไฟเข้าที่ขดลวดอยู่ 4 จังหวะต่อการหมุน 1 รอบ



$\text{delay} \propto \frac{1}{V}$

90°

การควบคุมการหมุนแบบ Wave Drive จะเป็นการป้อนกระแสไฟให้กับขดลวดของสเต็ปเปอร์มอเตอร์ทีละขด โดยจะป้อนกระแสเรียงตามลำดับกันไป ดังนั้นกระแสที่ไหลในขดลวดจะไหลในทิศทางเดียวกันทุกขด การควบคุมแบบนี้ทำได้ง่ายแต่แรงขับของสเต็ปเปอร์มอเตอร์ที่ได้มีน้อย ความเร็วที่ได้จากการหมุนของสเต็ปเปอร์มอเตอร์จะขึ้นอยู่กับการหน่วงเวลา (Time Delay) ของการป้อนกระแสไฟให้กับขดลวดในแต่ละครั้งตามลำดับ ถ้า Time Delay มีค่ามากมอเตอร์จะหมุนช้า และถ้า Time Delay มีค่าน้อยมอเตอร์จะหมุนเร็ว แต่ค่าน้อยมากก็อาจจะไม่เสถียรได้

- ✓ 3. การทดลองการทำงาน จากโปรแกรมในข้อ 12 ให้เพิ่มโปรแกรมการส่งข้อมูลไปที่ Stepping Motor โดยใช้การควบคุมแบบ Wavedrive เพื่อจ่ายไฟให้ทำงานครึ่งละ 1 ขด ซึ่งก็คือให้ทำงานครึ่งละ 1 เฟส ทดลองการทำงานของโปรแกรมและให้บันทึกผลที่ได้

```

void wavedrive()
{
    // 1
    digitalWrite(motorPin4, HIGH);
    digitalWrite(motorPin3, LOW);
    digitalWrite(motorPin2, LOW);
    digitalWrite(motorPin1, LOW);
    delay(motorSpeed);
    // 2
    digitalWrite(motorPin4, LOW);
    digitalWrite(motorPin3, HIGH);
    digitalWrite(motorPin2, LOW);
    digitalWrite(motorPin1, LOW);
    delay(motorSpeed);
}

```

```

// 3
digitalWrite(motorPin4, LOW);
digitalWrite(motorPin3, LOW);
digitalWrite(motorPin2, HIGH);
digitalWrite(motorPin1, LOW);
delay(motorSpeed);
// 4
digitalWrite(motorPin4, LOW);
digitalWrite(motorPin3, LOW);
digitalWrite(motorPin2, LOW);
digitalWrite(motorPin1, HIGH);
delay(motorSpeed);
}

```

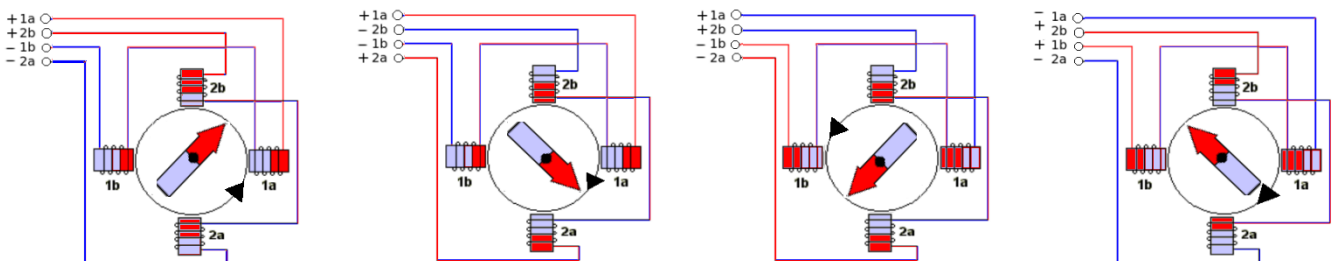
$$\frac{\text{องศา}}{\text{step}} = \frac{360^\circ}{515} = 0.7$$

4. Stepping Motor ทำงานมีค่ามุมหมุนต่อ Step เท่ากับ 0.7 องศา
5. ถ้าต้องการให้หมุน 1 รอบจะต้องใช้ทั้งหมดเท่ากับ 515 Step
6. การหมุนของแต่ละ Step ในโปรแกรมเป็นการหมุนตามเข็มนาฬิกาหรือทวนเข็มนาฬิกา ตามเข็มนาฬิกา
7. ถ้าต้องการให้หมุนในทิศทางตรงกันข้ามกันต้องแก้ไขโปรแกรมในส่วนไหน

แก้ไขลำดับการจ่ายไฟในขดลวดในฟังก์ชัน wave-drive จาก 4-3-2-1 เป็น 1-2-3-4

8. ให้ทดลองแก้ไขค่าใน Delay ให้น้อยลงและมากขึ้นและอธิบายผลลัพธ์ที่ได้

เพื่อลด delay ให้น้อยลง step จะเพิ่มเร็วขึ้น , step motor หมุนเร็วขึ้น
เพื่อเพิ่ม delay ในมากขึ้น step จะเพิ่มช้าลง , step motor หมุนช้าลง



30 การควบคุมการหมุนแบบ Full Step จะเป็นการป้อนกระแสไฟให้กับขดลวดของสเต็ปเปอร์มอเตอร์แบบทีละ 2 เฟสพร้อมกัน โดยต้องป้อนกระแสเรียงตามลำดับกันไปครั้งละ 2 ขด ดังนั้นจึงมีกระแสไหลในขดลวดของมอเตอร์มากขึ้น ซึ่งทำให้มอเตอร์มีแรงบิดในการหมุนมากขึ้นตามไปด้วย

9. จากโปรแกรมในข้อ 13 ให้เพิ่มโปรแกรมการส่งข้อมูลไปที่ Stepping Motor โดยใช้การควบคุมแบบ Fullstep ให้ทดลองการทำงานของโปรแกรมและบันทึกผลที่ได้

```

void fullstep()
{
    // 1
    digitalWrite(motorPin4, HIGH);
    digitalWrite(motorPin3, HIGH);
    digitalWrite(motorPin2, LOW);
    digitalWrite(motorPin1, LOW);
    delay (motorSpeed);
}

```

```

// 2
digitalWrite(motorPin4, LOW);
digitalWrite(motorPin3, HIGH);
digitalWrite(motorPin2, HIGH);
digitalWrite(motorPin1, LOW);
delay(motorSpeed);
// 3
digitalWrite(motorPin4, LOW);
digitalWrite(motorPin3, LOW);
digitalWrite(motorPin2, HIGH);
digitalWrite(motorPin1, HIGH);
delay (motorSpeed);
// 4
digitalWrite(motorPin4, HIGH);
digitalWrite(motorPin3, LOW);
digitalWrite(motorPin2, LOW);
digitalWrite(motorPin1, HIGH);
delay(motorSpeed);
}

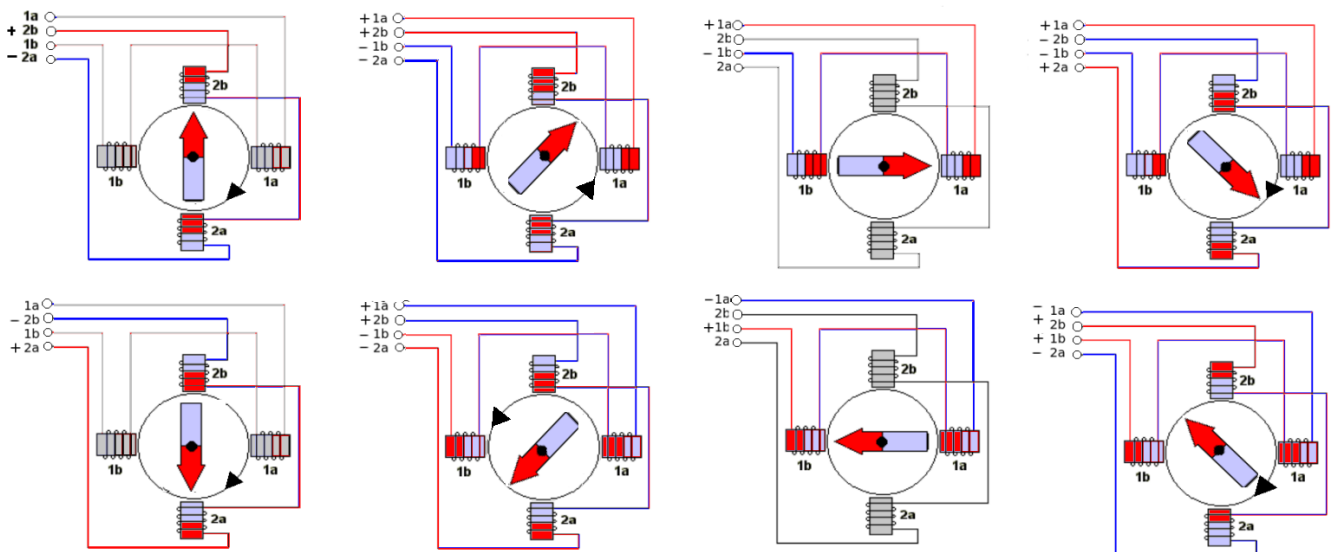
```

10. Stepping Motor ทำงานแบบ Full Step มีมุมหมุนต่อ Step เท่ากับแบบ Wavedrive หรือไม่

มุมหมุนเท่ากัน (๖๐°)

11. การทำงานในแต่ละ Step ของแบบ Full Step กับแบบ Wavedrive อยู่ในตำแหน่งสอดคล้องกันหรือไม่

ไม่ Full Step และ Wave Drive จะต่างกันอยู่ 45°



45° การควบคุมการหมุนแบบ Half Step เป็นการป้อนกระแสไฟให้กับขดลวดของสเต็ปเปอร์มอเตอร์ครั้งละ 1 เฟส และ 2 เฟส สลับกันไป ทำให้สเต็ปเปอร์มอเตอร์มีความละเอียดของตำแหน่งในการหมุนเพิ่มขึ้น 2 เท่า ซึ่งจะหมุนได้ครั้งละครึ่งสเต็ป โดยที่ไม่ต้องปรับเปลี่ยนฮาร์ดแวร์เพียงแต่แก้ไขโปรแกรมวิธีการจ่ายกระแสไฟเข้าขดลวดให้เพิ่มมากขึ้น

12. จากโปรแกรมในข้อ 19 ให้เพิ่มโปรแกรมการส่งข้อมูลไปที่ Stepping Motor ด้วยวิธีการควบคุมแบบ Half Step

โดยนำข้อมูลของทั้งสองแบบมารวมกัน ให้ทดลองการทำงานของโปรแกรมและบันทึกผลที่ได้

```

void halfstep()
{
// 1
digitalWrite(motorPin4, HIGH);
digitalWrite(motorPin3, LOW);
digitalWrite(motorPin2, LOW);
digitalWrite(motorPin1, LOW);
delay(motorSpeed);
}

```

```

// 2
digitalWrite(motorPin4, HIGH);
digitalWrite(motorPin3, HIGH);
digitalWrite(motorPin2, LOW);
digitalWrite(motorPin1, LOW);
delay (motorSpeed);
// 3
digitalWrite(motorPin4, LOW);
digitalWrite(motorPin3, HIGH);
digitalWrite(motorPin2, LOW);
digitalWrite(motorPin1, LOW);
delay(motorSpeed);
// 4
digitalWrite(motorPin4, LOW);
digitalWrite(motorPin3, HIGH);
digitalWrite(motorPin2, HIGH);
digitalWrite(motorPin1, LOW);
delay(motorSpeed);
// 5
digitalWrite(motorPin4, LOW);
digitalWrite(motorPin3, LOW);
digitalWrite(motorPin2, HIGH);
digitalWrite(motorPin1, LOW);
delay(motorSpeed);
// 6
digitalWrite(motorPin4, LOW);
digitalWrite(motorPin3, LOW);
digitalWrite(motorPin2, HIGH);
digitalWrite(motorPin1, HIGH);
delay (motorSpeed);
// 7
digitalWrite(motorPin4, LOW);
digitalWrite(motorPin3, LOW);
digitalWrite(motorPin2, LOW);
digitalWrite(motorPin1, HIGH);
delay(motorSpeed);
// 8
digitalWrite(motorPin4, HIGH);
digitalWrite(motorPin3, LOW);
digitalWrite(motorPin2, LOW);
digitalWrite(motorPin1, HIGH);
delay(motorSpeed);
}

```

$$\frac{\text{องศา}}{\text{step}} = \frac{360}{254} = 1.4$$

13. Stepping Motor ทำงานแบบ Half Step มีมุมหมุนต่อStep เท่ากับ 1.4 องศา
14. ถ้าต้องการให้แบบ Half Step หมุน 1 รอบจะต้องใช้ทั้งหมดเท่ากับ 254 Step
15. ให้แก้ไขโปรแกรมโดยเพิ่มสวิทช์ 1 ตัว กำหนดว่าเมื่อมีการกดสวิทช์ให้ Stepping Motor หมุนตามเข็มนาฬิกา และถ้าปล่อยสวิทช์ให้ Stepping Motor หมุนทวนเข็มนาฬิกา