**Lab# 7**

การใช้งาน Stepping Motor

**สเต็ปมอเตอร์ (Step motor)**

สเต็ปมอเตอร์ เป็นมอเตอร์ประเภทหนึ่งที่หมุนได้องศาเพียงเล็กน้อยตามแรงดันไฟฟ้าที่ป้อนเป็นพัลซ์ ซึ่งต่าง จากมอเตอร์ไฟฟ้าทั่วไปที่จะหมุนทันทีและตลอดเวลาเมื่อป้อนแรงดันไฟฟ้าข้อดีของสเต็ปมอเตอร์ คือ สามารถกำหนดตำแหน่งของการหมุน (องศา หรือระยะทาง) ได้อย่างละเอียด โดยใช้คอมพิวเตอร์หรือไมโครคอนโทรลเลอร์เป็นตัวควบคุมการทำงาน

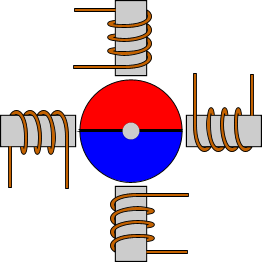
โครงสร้างของสเต็ปมอเตอร์ ประกอบด้วยขั้ว แม่เหล็กบนสเตเตอร์ทำมาจากแผ่นเหล็กวงแหวนที่มีซี่ยื่นออกมาประกอบกันเป็นชั้น ๆ โดยที่แต่ละซี่นั้น จะมีคอยล์(ขดลวด)พันสวมอยู่ เมื่อมีการป้อนกระแสผ่านคอยล์ทำให้เกิดสนามแม่เหล็กไฟฟ้า(Electromagnetic) ดังรูป แสดงให้เห็นองค์ประกอบต่างๆ ของสเต็ปมอเตอร์ ในที่นี้ถ้าเพิ่มจำนวนของขั้ว แม่เหล็กมากขึ้น จะสามารถเพิ่มจำนวนของสเต็ปต่อวงจรรอบมากขึ้น ตามไปด้วย



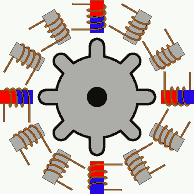
แสดงโครงสร้างและส่วนประกอบของสเต็ปมอเตอร์

**ประเภทของสเต็ปมอเตอร์**

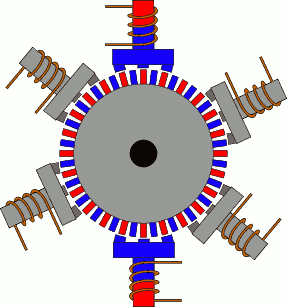
1.แบบแม่เหล็กถาวร (Permanent Magnet-PM) คือ สเต็ปมอเตอร์แบบ PM จะมีสเตเตอร์(Stator) ที่พันขดลวดไว้หลายๆ โพล โดยมีโรเตอร์ (Rotor) เป็นรูปทรงกระบอกฟันเลื่อย และโรเตอร์ทำด้วยแม่เหล็กถาวร เพื่อป้อนไฟกระแสตรงให้กับขดสเตเตอร์ จะทำให้เกิดแรงแม่เหล็กไฟฟ้าผลักต่อโรเตอร์ ทำให้มอเตอร์หมุนมอเตอร์แบบ PM จะเกิดแรงฉุดยึดให้โรเตอร์หยุดอยู่กับที่ แม้จะไม่ได้ป้อนไฟเข้าขดลวด



2.แบบแปรค่ารีลักแตนซ์ (Variable Reluctance - VR) คือ สเต็ปมอเตอร์แบบ VR จะมีการหมุนโรเตอร์ได้อย่างอิสระ แม้จะไม่ได้จ่ายไฟให้โรเตอร์ทำจากสารเฟอร์โรแมกเนติก กำลังอ่อน มีลักษณะเป็นฟันเลื่อย รูปทรงกระบอกโดยจะมีความสัมพันธ์ โดยตรงกับจำนวนโพลในสเตเตอร์แรงบิดที่เกิดขึน้ จะไปหมุนโรเตอร์ ไปในเส้นทางของอำนาจแม่เหล็กที่มีค่ารีลักแตนท์ต่ำที่สุดตำแหน่งที่จะเกิดแน่นอนและมีเสถียรภาพแต่จะเกิดขึ้น ได้หลายๆ จุดดังนั้น เมื่อป้อนไฟเข้าขดลวดต่างๆ ในมอเตอร์แตกต่างขดกันไป ก็ทำให้มอเตอร์ หมุนไปตำแหน่งต่างๆ กันโรเตอร์ของ VR จะมีความเฉื่อยของโรเตอร์น้อยจึงมีความเร็วรอบสูงกว่ามอเตอร์แบบ PM



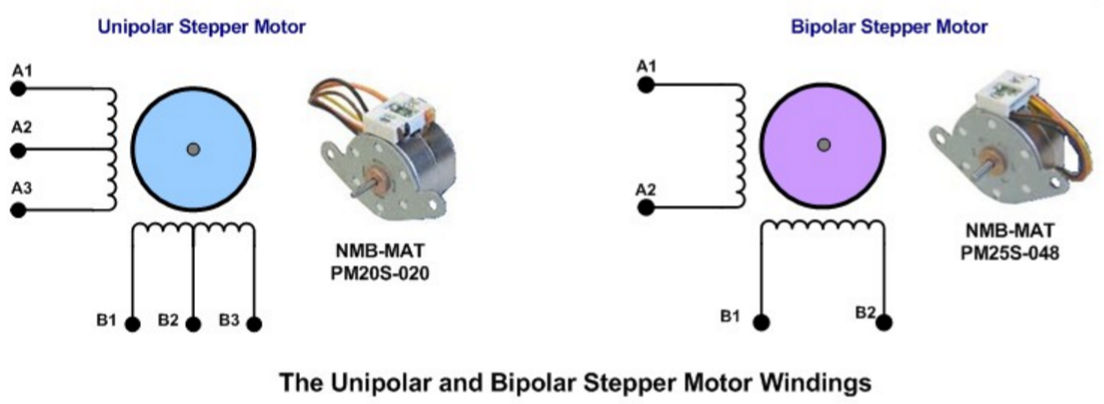
3.แบบผสม (Hybrid-H) คือ สเต็ปมอเตอร์แบบ H จะเป็นลูกผสมของ VR กับ PM โดยจะมีสเตเตอร์คล้ายกับที่ใช้ใน VR โรเตอร์มีหมวกหุ้ม ปลายซึ่งมีลักษณะของสารแม่เหล็กที่มีกำลังสูงโดยการควบคุมขนาดรูปร่างของหมวกแม่เหล็กอย่างดีทำให้ได้มุม การหมุนและครั้งน้อยและแม่นยำ ข้อดีก็คือ ให้แรงบิดสูงและมีขนาดกระทัดรัด และให้แรงฉุดยึดโรเตอร์นิ่งกับที่ตอนไม่จ่ายไฟ



**การพันขดลวดบนสเตเตอร์ของสเต็ปมอเตอร์สามารถแบ่งได้ดังนี้**

1. แบบไบโพลาร์ (Bipolar) คือ การพันขดลวดหนึ่งขดจะกี่รอบก็ตามแต่ โดยทั่วไปจะแยกขดลวดออกจากกันเป็น 2 คู่ชัดเจน ในแต่ละขั้วแม่เหล็กของสเตเตอร์ ขั้วแม่เหล็กที่เกิดขึ้นที่สเตเตอร์จะถูกกำหนดโดยทิศทางของการไหลของกระแสไฟฟ้า ซึ่งสามารถทำให้เกิดขั้ว แม่เหล็กในทิศทางตรงกันข้ามได้ เมื่อทำการกลับทิศทางของการไหลในกระแสไฟฟ้า โดยมาจากการควบคุมของวงจรสวิทชิ่งให้กลับขั้วไฟฟ้า โดยจะเรียกสเต็ปมอเตอร์ประเภทนี้อีกชื่อหนึ่งว่าสเต็ปมอเตอร์ 2 เฟส (2 Phase) ซึ่งจะมี 4 เส้น

2. แบบยูนิโพลาร์ (Unipolar) คือ การพันขดลวดชนิดนี้จะคล้ายกันกับกรณีแรก แตกต่างกันตรงที่จะมีการเชื่อมจุดศูนย์กลางของแต่ละขดเข้าหากัน ในแต่ละขั้วแม่เหล็กของสเตเตอร์ขั้วแม่เหล็กที่เกิดขึ้นที่สเตเตอร์จะถูกกำหนดโดยทิศทางของการไหลของกระแสไฟฟ้าเช่นเดียวกัน และสามารถกลับทิศทางขั้วแม่เหล็กได้จากการกลับทิศทางการไหลของกระแสไฟฟ้า สเต็ปมอเตอร์ชนิดนี้เรียกอีกชื่อหนึ่งว่า สเต็ปมอเตอร์ 5 เฟส (5 Phase) โดยทั่วไปจะมี 5 หรือ 6 เส้น



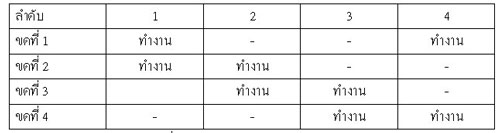
**การควบคุมการหมุนของสเต็ปมอเตอร์**

การควบคุมและสั่งงานให้สเต็ปมอเตอร์ทำงาน ไปทีละสเต็ปสามารถทำได้โดยการจ่ายกำลังไฟไปยังขดลวด ในแต่ละขอบนสเตเตอร์ โดยการป้อนจะทำในลักษณะเป็นลำดับหรือเรียกว่าซีเควนเวียลในลูปที่ถูกต้อง ซึ่งจะแบบ ได้เป็น 3 รูปแแบ คือ แบบเวฟ (Wave) และแบบ 2 เฟส (2 Phase) และแบบครึ่งสเต็ป (Half step) ทั้ง 3 แบบนี้ก็จะมีข้อดีและข้อเสียต่างกันออกไป

1.**แบบเวฟ (Wave)** จะเป็นการกระตุ้นแบบที่ง่ายที่สุด ซึ่งจะทำการกระตุ้นขดลวดทีละขดหรือ 1 เฟส ในเวลาหนึ่งๆ เรียงกันไป ตัวอย่างเช่น ขดที่ 1 , 2 , 3 , 4 ,1, 2 , 3 , 4 เป็นลำดับอย่างนี้หรือ ขด 4 , 3 , 2 , 1 , 4 , 3 , 2 ,1 เป็นลำดับกันไป ทั้งนี้ขึ้น อยู่กับทิศทางที่เราต้องให้มอเตอร์หมุนไป วงจรที่นำมากระตุ้นนั้น จะมีราคาค่อนข้างจะถูกกว่าและง่ายกว่า



2.**แบบ 2 เฟส (2 Phase)** จะคล้ายกับการกระตุ้นในแบบเวฟแต่จะต่างกันตรงที่แบบ 2 เฟส จะกระตุ้นทีละ 2 ขด ที่อยู่ใกล้กันใน เวลาเดียวกันและจะเรียงลำดับกันไป ดังเช่นแบบเดียวกับแบบเวฟ จะยกตัวอย่างการกระตุ้นขดลวดในลักษณะ รอบการวนลูปได้ดังนี้ 12, 23, 34, 41, 12, 23, 34, 41 เรียงลำดับกันไปเรื่อยๆ หรือจะเป็น 43, 32, 21,14, 43, 32, 21,14 เรียงกันไปเรื่อยๆเช่นกัน ถ้าจะกล่าวถึงข้อดีข้อเสียของแบบ 2 เฟส มีดังนี้ ข้อดีคือ การที่จะเพิ่มจำนวนขดลวดที่ถูกกระตุ้นจะทำให้แรงบิดได้มากกว่า แบบเวฟ ซึ่งโรเตอร์จะหมุนด้วยแรง ดึงแบบเต็มๆแรงจากทั้ง 2 ขดลวดที่กระตุ้นพร้อมกัน และข้อเสียของแบบ 2 เฟส คือการกระตุ้นขดลวดนั้น ต้องใช้กำลังไฟมากขึ้น เป็น 2 เท่าของแบบเวฟ

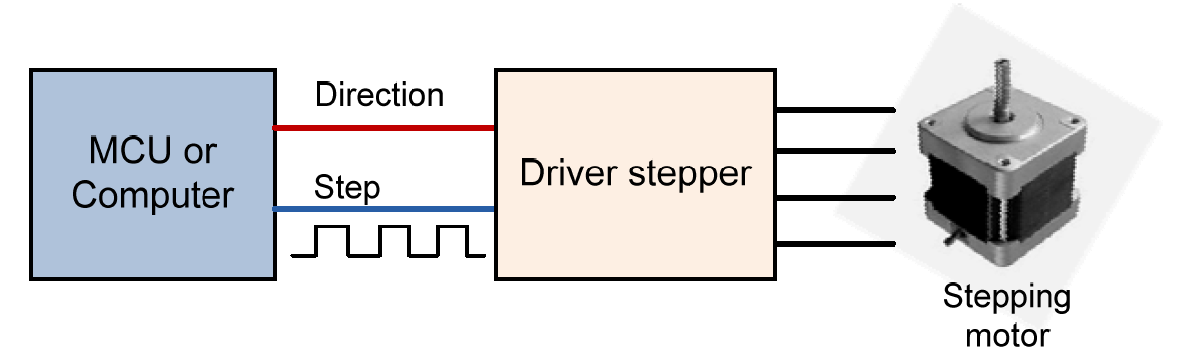


3.**แบบครึ่งสเต็ป (Half step)** การควบคุมแบบครึ่งสเต็ปเป็นการรวมเอาทั้ง 2 แบบข้างต้นมารวมกัน โดยมีรอบการวนลูปดังนีคื้อ 1, 12, 2, 23, 3, 34, 4, 41, … ไปเรื่อยๆ หรือจะเป็น 14, 4, 43, 3, 32, 2, 21, 1, … ไปเรื่อยๆ ข้อดีคือให้แรงบิดที่สูงและความละเอียดมากด้วย

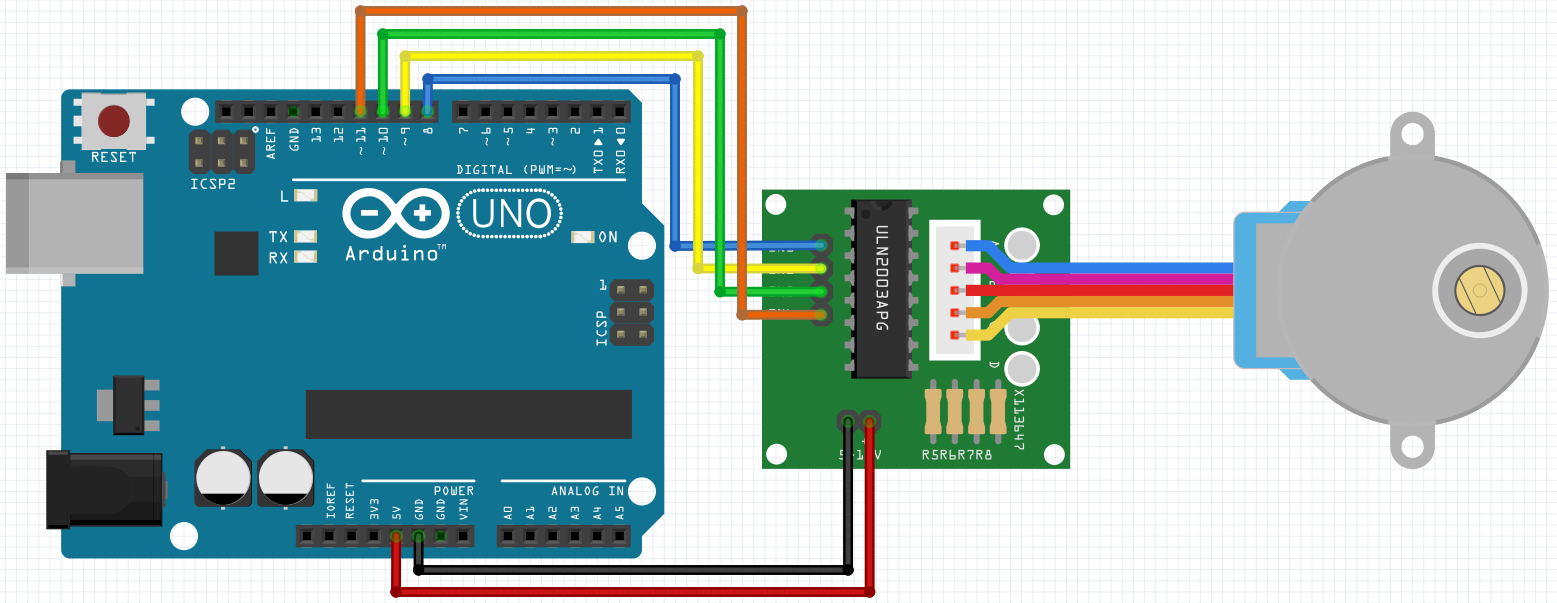


**ไดรเวอร์สเต็ปมอเตอร์ (Driver Step Motor)**

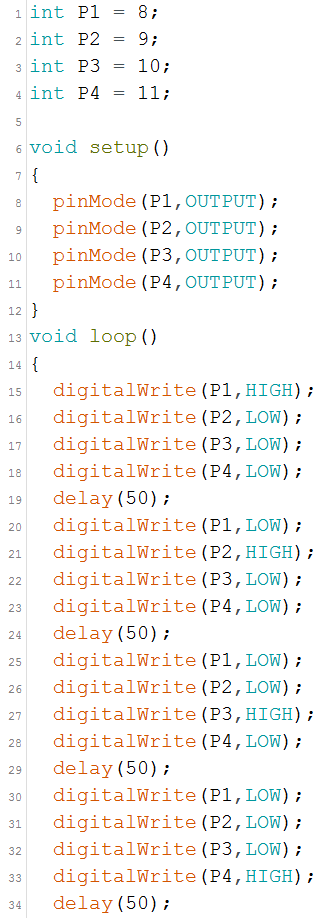
จากหลักการควบคุมการหมุนของสเต็ปมอเตอร์จากที่ได้กล่าวมาก่อนหน้านี้จะพบว่าการควบคุมการหมุนของสเต็ปมอเตอร์ถือว่ามีความลำบากในเรื่องของจำนวนสายสัญญาณพอสมควรดังนั้น ได้มีการพัฒนาวงจรหรืออุปกรณ์อิเล็กทรอนิกส์ขึ้น มาเพื่อช่วยให้ง่ายและมีความสะดวกต่อการนำไปควบคุมการทำงานของสเต็ปมอเตอร์มายิ่งขึ้น โดยเรียกอุปกรณ์ประเภทนี้ว่า ไดรเวอร์ สเต็ปมอเตอร์ (Driver Step Motor) โดยการทำงานของไดรเวอร์จะรับสัญญาณจากคอมพิวเตอร์หรือไมโครคอนโทรเลอร์เพื่อไปควบคุมการหมุนของมอเตอร์ แสดงหลักการดังรูป



วิธีการขั้น พื้น ฐานที่สุดในการควบคุมสเต็ปมอเตอร์คือการจ่ายสัญญาณ 2 อย่างคือ 1.สัญญาณสเต็ป (Step) มีลักษณะแบบเดียวกันกับสัญญาณนาฬิกา ทำหน้าที่ในการกำหนดจำนวนสเต็ปหรือจำนวนรอบการหมุนของสเต็ปมอเตอร์ว่าต้องการให้หมุนไปกี่องศาหรือกี่รอบและความถี่ของสัญญาณสเต็ปจะเป็นตัวกำหนดความเร็วการหมุนของตัวมอเตอร์ด้วย และ 2.สัญญาณทิศทาง (Direction) คือสัญญาณที่เป็นตัวกำหนดทิศทางการหมุนของสเต็ปมอเตอร์ว่าจะให้หมุนตามหรือทวนเข็มนาฬิกา โดยทั่วไปจะกำหนดได้จากลอจิก “1” (HIGH) หรือ “0” (LOW) นั่นเอง

**ตัวอย่างการต่อวงจร และ Source Code การใช้งาน Stepping Motor แบบเวฟ (Wave) หมุนตามเข็มนาฬิกา**

Source Code



**การทดลองที่ 1 จงเขียนโปรแกรมเพื่อควบคุมการหมุนของ Stepping Motor แบบตามเข็ม และทวนเข็มนาฬิกา แบบเวฟ (Wave)**

Source Code

int P1 = 8;

int P2 = 9;

int P3 = 10;

int P4 = 11;

void setup() {

pinMode(P1,OUTPUT);

pinMode(P2,OUTPUT);

pinMode(P3,OUTPUT);

pinMode(P4,OUTPUT);

}

void loop() {

for(int i = 1;i < 512;i++)

{

digitalWrite(P1,HIGH);

digitalWrite(P2,LOW);

digitalWrite(P3,LOW);

digitalWrite(P4,LOW);

delay(50);

digitalWrite(P1,LOW);

digitalWrite(P2,HIGH);

digitalWrite(P3,LOW);

digitalWrite(P4,LOW);

delay(50);

digitalWrite(P1,LOW);

digitalWrite(P2,LOW);

digitalWrite(P3,HIGH);

digitalWrite(P4,LOW);

delay(50);

digitalWrite(P1,LOW);

digitalWrite(P2,LOW);

digitalWrite(P3,LOW);

digitalWrite(P4,HIGH);

delay(50);

}

for(int i = 1;i < 512;i++)

{

digitalWrite(P1,LOW);

digitalWrite(P2,LOW);

digitalWrite(P3,HIGH);

digitalWrite(P4,LOW);

delay(50);

digitalWrite(P1,LOW);

digitalWrite(P2,HIGH);

digitalWrite(P3,LOW);

digitalWrite(P4,LOW);

delay(50);

digitalWrite(P1,HIGH);

digitalWrite(P2,LOW);

digitalWrite(P3,LOW);

digitalWrite(P4,LOW);

delay(50);

}

}

บันทึกผลการทดลอง

หมุน step motor ไปหนึ่งรอบตามเข็มนาฬิกา เมื่อครบรอบและหมุนกลับ

**การทดลองที่ 2 จงเขียนโปรแกรมเพื่อควบคุมการหมุนของ Stepping Motor แบบตามเข็ม และทวนเข็มนาฬิกา แบบ 2 เฟส (2 Phase)**

Source Code

int P1 = 8;

int P2 = 9;

int P3 = 10;

int P4 = 11;

void setup() {

pinMode(P1,OUTPUT);

pinMode(P2,OUTPUT);

pinMode(P3,OUTPUT);

pinMode(P4,OUTPUT);

}

void loop() {

for(int i = 1;i < 512;i++)

{

digitalWrite(P1,HIGH);

digitalWrite(P2,HIGH);

digitalWrite(P3,LOW);

digitalWrite(P4,LOW);

delay(50);

digitalWrite(P1,LOW);

digitalWrite(P2,HIGH);

digitalWrite(P3,HIGH);

digitalWrite(P4,LOW);

delay(50);

digitalWrite(P1,LOW);

digitalWrite(P2,LOW);

digitalWrite(P3,HIGH);

digitalWrite(P4,HIGH);

delay(50);

digitalWrite(P1,HIGH);

digitalWrite(P2,LOW);

digitalWrite(P3,LOW);

digitalWrite(P4,HIGH);

delay(50);

}

for(int i = 1;i < 512;i++)

{

digitalWrite(P1,LOW);

digitalWrite(P2,LOW);

digitalWrite(P3,HIGH);

digitalWrite(P4,HIGH);

delay(50);

digitalWrite(P1,LOW);

digitalWrite(P2,HIGH);

digitalWrite(P3,HIGH);

digitalWrite(P4,LOW);

delay(50);

digitalWrite(P1,HIGH);

digitalWrite(P2,HIGH);

digitalWrite(P3,LOW);

digitalWrite(P4,LOW);

delay(50);

}

}

บันทึกผลการทดลอง

หมุน step motor ไปหนึ่งรอบตามเข็มนาฬิกา เมื่อครบรอบและหมุนกลับ แต่ แรงบิด เป็น 2 เท่ากินไฟเพิ่มเป็น 2 เท่า

**การทดลองที่ 3 จงเขียนโปรแกรมเพื่อควบคุมการหมุนของ Stepping Motor แบบตามเข็ม และทวนเข็มนาฬิกา แบบครึ่งสเต็ป (Half step)**

Source Code

int P1 = 8;

int P2 = 9;

int P3 = 10;

int P4 = 11;

void setup() {

pinMode(P1,OUTPUT);

pinMode(P2,OUTPUT);

pinMode(P3,OUTPUT);

pinMode(P4,OUTPUT);

}

void loop() {

for(int i = 1;i < 512;i++)

{

digitalWrite(P1,HIGH);

digitalWrite(P2,LOW);

digitalWrite(P3,LOW);

digitalWrite(P4,LOW);

delay(50);

digitalWrite(P1,HIGH);

digitalWrite(P2,HIGH);

digitalWrite(P3,LOW);

digitalWrite(P4,LOW);

delay(50);

digitalWrite(P1,LOW);

digitalWrite(P2,HIGH);

digitalWrite(P3,LOW);

digitalWrite(P4,LOW);

delay(50);

digitalWrite(P1,LOW);

digitalWrite(P2,HIGH);

digitalWrite(P3,HIGH);

digitalWrite(P4,LOW);

delay(50);

digitalWrite(P1,LOW);

digitalWrite(P2,LOW);

digitalWrite(P3,HIGH);

digitalWrite(P4,LOW);

delay(50);

digitalWrite(P1,LOW);

digitalWrite(P2,LOW);

digitalWrite(P3,HIGH);

digitalWrite(P4,HIGH);

delay(50);

digitalWrite(P1,LOW);

digitalWrite(P2,LOW);

digitalWrite(P3,LOW);

digitalWrite(P4,HIGH);

delay(50);

digitalWrite(P1,HIGH);

digitalWrite(P2,LOW);

digitalWrite(P3,LOW);

digitalWrite(P4,HIGH);

delay(50);

}

for(int i = 1;i < 512;i++)

{

digitalWrite(P1,LOW);

digitalWrite(P2,LOW);

digitalWrite(P3,LOW);

digitalWrite(P4,HIGH);

delay(50);

digitalWrite(P1,LOW);

digitalWrite(P2,LOW);

digitalWrite(P3,HIGH);

digitalWrite(P4,HIGH);

delay(50);

digitalWrite(P1,LOW);

digitalWrite(P2,LOW);

digitalWrite(P3,HIGH);

digitalWrite(P4,LOW);

delay(50);

digitalWrite(P1,LOW);

digitalWrite(P2,HIGH);

digitalWrite(P3,HIGH);

digitalWrite(P4,LOW);

delay(50);

digitalWrite(P1,LOW);

digitalWrite(P2,HIGH);

digitalWrite(P3,LOW);

digitalWrite(P4,LOW);

delay(50);

digitalWrite(P1,HIGH);

digitalWrite(P2,HIGH);

digitalWrite(P3,LOW);

digitalWrite(P4,LOW);

delay(50);

digitalWrite(P1,HIGH);

digitalWrite(P2,LOW);

digitalWrite(P3,LOW);

digitalWrite(P4,LOW);

delay(50);

digitalWrite(P1,HIGH);

digitalWrite(P2,LOW);

digitalWrite(P3,LOW);

digitalWrite(P4,HIGH);

delay(50);

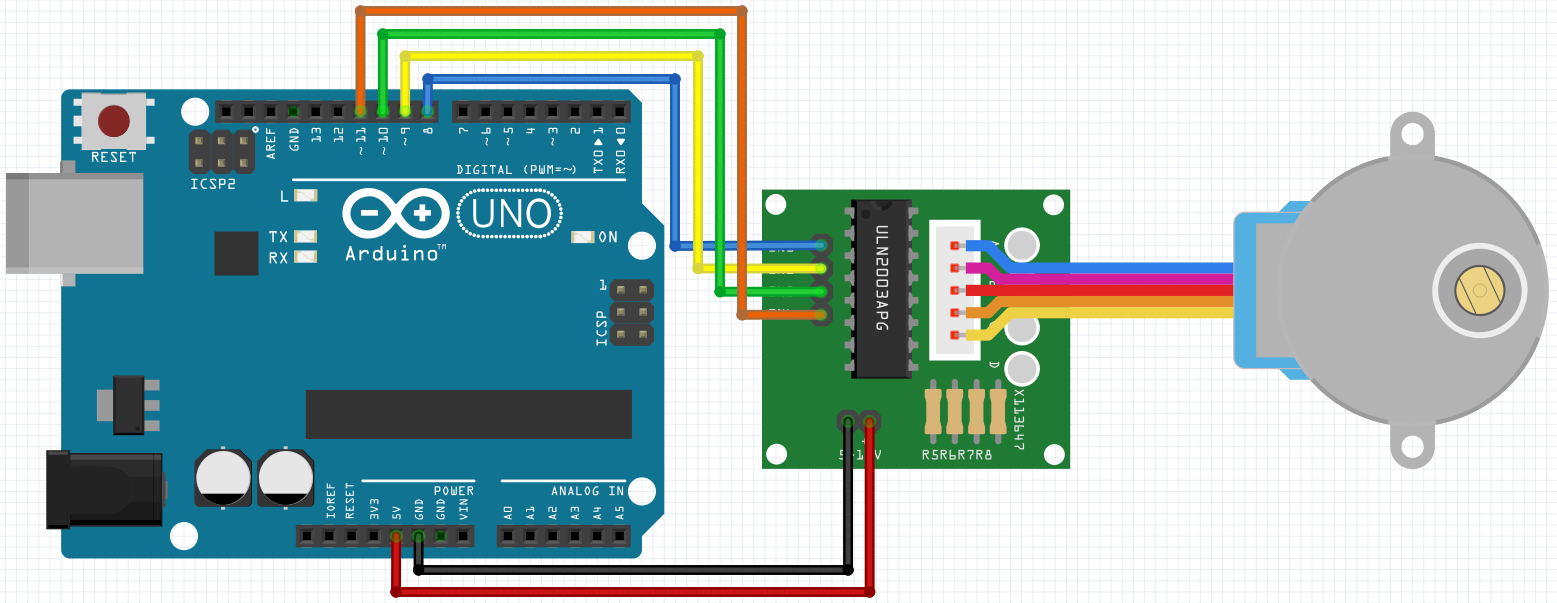
}

}

บันทึกผลการทดลอง

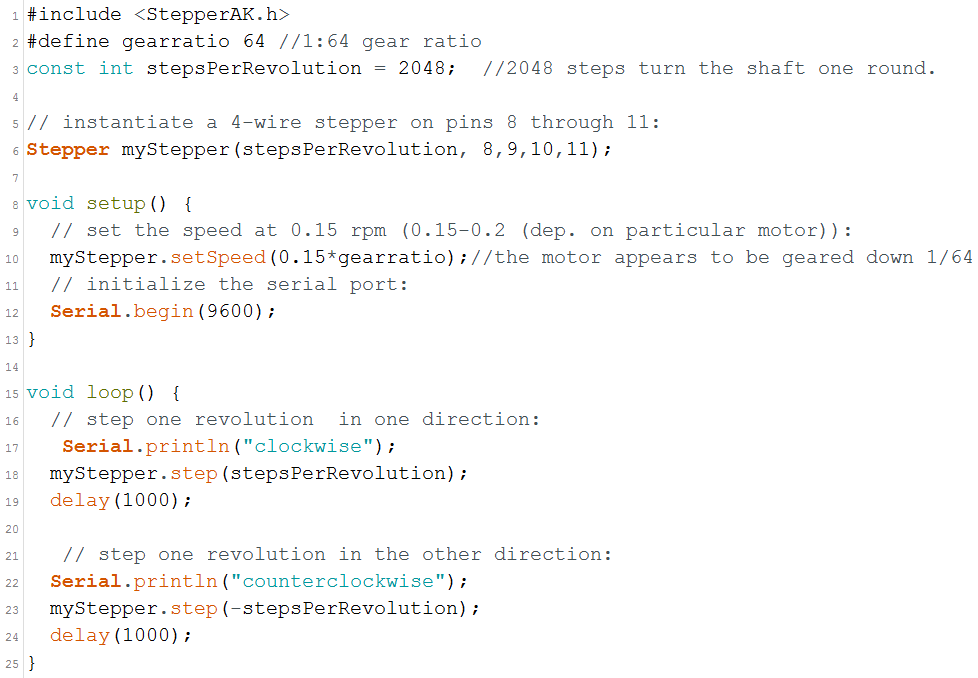
หมุน step motor ไปหนึ่งรอบตามเข็มนาฬิกา เมื่อครบรอบและหมุนกลับ

**ตัวอย่างการต่อวงจร และ Source Code การใช้งาน Stepping Motor แบบใช้งาน Library**

รูปแบบการต่อวงจร

Download library [stepperarduinokitlibrary.zip](https://sites.google.com/a/ku.th/weeraphan/classroom/embedded-system/stepperarduinokitlibrary.zip?attredirects=0&d=1)

Source Code



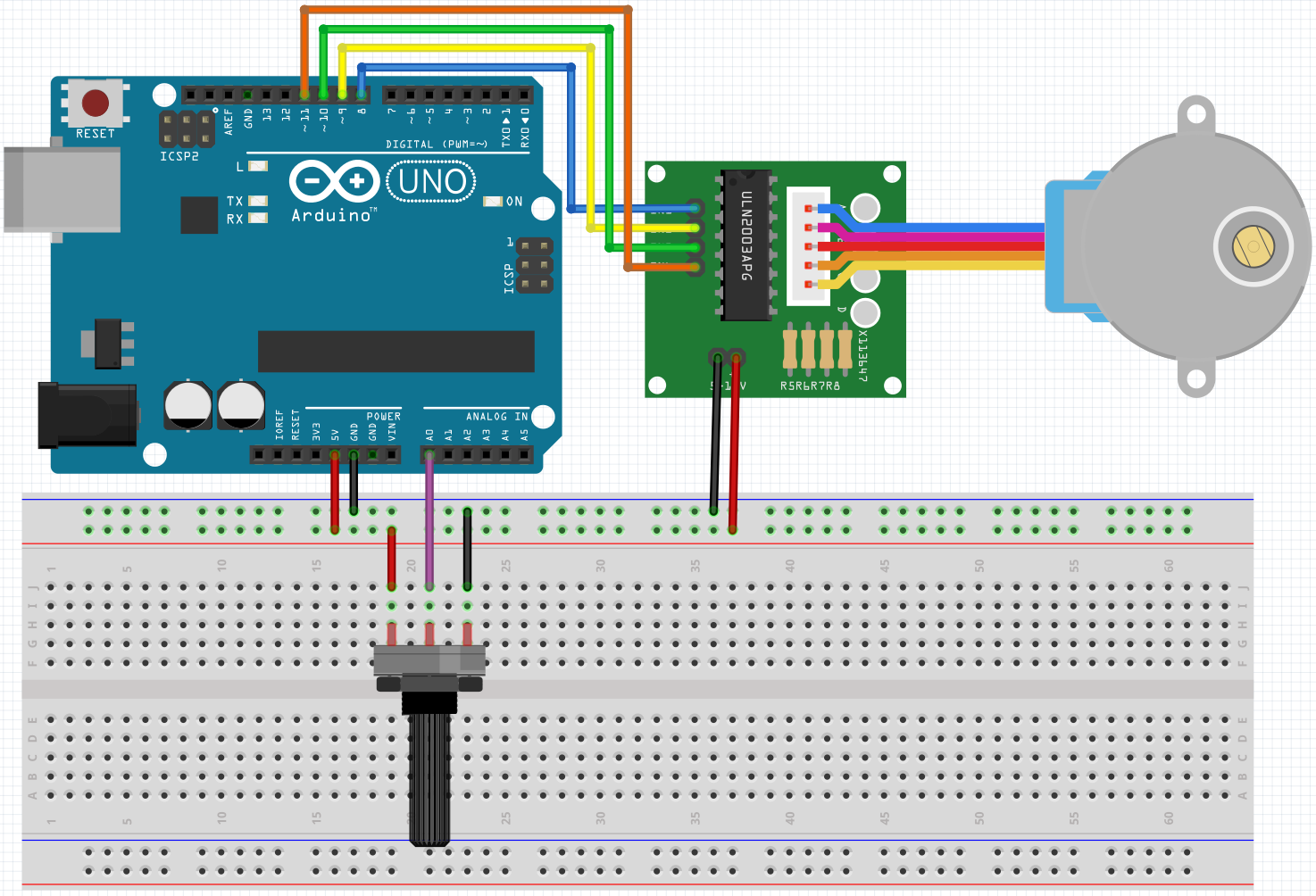
บันทึกผลการทดลอง

หมุนตามเข็มนาฬิกา แล้ว delay 1 วินาที แล้วหมุนย้อนกลับ

**การทดลองที่ 4 รับค่าจากตัวต้านทานปรับค่าได้ เพื่อควบคุมการหมุนของ Stepping Motor**

ในการทดลองนี้ ใช้การรับค่าจากตัวต้านทางปรับค่าได้ VR และนำสัญญาณที่ได้มาควมคุมการหมุนของ Stepping Motor โดยการหมุนตัวต้านทานปรับค่าได้ โดยค่าที่ได้คือ 0 = 0 องศา และ 1023 = 360 องศา เพื่อให้ Stepping Motor หมุนตามองศาที่ปรับได้จากตัวต้านทาน

รูปแบบการต่อวงจร



Source Code

#include <StepperAK.h>

#define gearratio 64

const int stepsPerRevolution = 2048;

Stepper myStepper(stepsPerRevolution,8,9,10,11);

int val;

int temp;

void setup() {

// put your setup code here, to run once:

myStepper.setSpeed(0.15\*gearratio);

temp = map(analogRead(0),0,1023,0,2048);

}

void loop() {

// put your main code here, to run repeatedly:

val = map(analogRead(0),0,1023,0,2048);

if(val < temp && (temp - val) >= 50){

myStepper.step(-(temp - val));

temp = val;

}else if(val > temp && (val - temp) >= 50){

myStepper.step((val - temp));

temp = val;

}

}

บันทึกผลการทดลอง

หมุน stepper motor ตามค่าที่อ่านได้จาก VR