**Lab# 8**

การใช้งาน Servo Motor

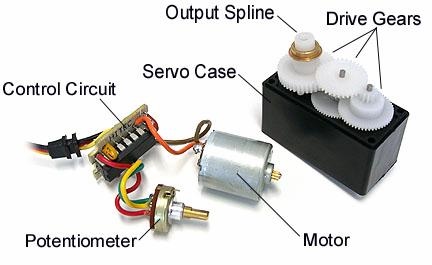
**เซอร์โวมอเตอร์ (Servo Motor)**

เซอร์โวมอเตอร์ เป็นมอเตอร์ที่ทำงานโดยใช้สัญญาณพัลส์ โดยภายในเซอร์โวมอเตอร์จะประกอบไปด้วย มอเตอร์ไฟฟ้ากระแสตรง ชุดเกียร์และส่วนควบคุม โดยจะประกอบกันอยู่ภายในชุดเดียวกัน ตัวเซอร์โวมอเตอร์จะมีสายสัญญาณ 3 เส้น คือ สายใช้งาน 1 เส้น อีก 2 เส้นจะเป็นสายสำหรับจ่ายไฟให้เซอร์โวมอเตอร์และสายสำหรับต่อลงกราวด์ ในการควบคุมเซอร์โวมอเอตร์นั้นจะทำให้หมุนไปทางซ้ายได้ 90 องศา ไปทางขวาได้ 90 องศา (180 องศา) และสามารถสั่งให้หมุนไปตามองศาที่กำหนดได้

การใช้งานเซอร์โวมอเตอร์นั้นจะนำไปใช้ในที่ที่ต้องการความแม่ยำในเรื่ององศา หรือ การหมุนไปตามองศาที่ต้องการ เช่น ใช้เป็นมอเตอร์บังคับการเลี้ยวของหางเสือ เรือ หรือ การเลี้ยวของเครื่องบินบังคับวิทยุ แม้แต่สร้างเป็นหุ่นยนต์เดินขนาดเล็ก เพราะตัวเซอร์โวมอเอตร์เองจะมีแรงบิดค่อนข้างสูง



**องค์ประกอบพื้นฐานของเซอร์โว**



องค์ประกอบหลักของเซอร์โวโดยทั่วไปแล้วจะมีส่วนประกอบหลักดังนี้คือ

1.Servo Case ซึ่งส่วนใหญ่จะทำมาจากพลาสติก

2.Motor ซึ่งเป็นส่วนให้กำลังในการหมุนของเซอร์โว

3.Control Circuit มีหน้าที่ในการถอดรหัสสัญญาณควบคุมจากรีซีฟซึ่งส่งมาเป็นแบบ PWM และส่งการควบคุมไปสั่งการทำงานของมอเตอร์ให้หมุนแขนของเซอร์โวให้อยู่ในตำแหน่งที่ได้ถอดรหัสมา

4.Potentiometer คือส่วนที่ตรวจวัดตำแหน่งของเซอร์โวและส่งสัญญาณกลับไปยัง Control Circuit เพื่อแก้ใขตำแหน่งให้ถูกต้องตามสัญญาณที่ได้เซ็ตไว้

5.Drive Gear คือชุดทดรอบจากการหมุนของมอเตอร์เพื่อให้ได้แรงบิดที่สูง

6.Output Spline คือส่วนที่ปัองกันการเสียดสีระหว่าง Servo Case และ Output shaft ซึ่งอาจใช้อุปกรณ์ประเภท Baring เพื่อช่วยลดแรงเสียดทานที่ดี

7.Servo wire คือสายไฟของเซอร์โวซึ่งมีรายละเอียดดังนี้

สายไฟของเซอร์โวจะมีอยู่สามเส้นซึ่งจะติดเป็นชุดเดียวกัน ซึ่งจะมีหน้าที่คือ

เส้นที่ 1 จ่ายไฟกระแส + DC ซึ่งแรงดันปรกติจะอยู่ที่ 5-6 โวลท์

เส้นที่ 2 เป็นสาย Ground หรือเป็นขั้ว – DC

เส้นที่ 3 เป็นสายสัญญาณ โดยที่รีซีฟจะส่งสัญญาณลักษณะ on/off pulsed

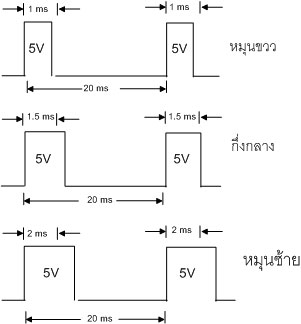
**การควบคุมทำงานของเซอร์โวมอเตอร์**

ในการควบคุมเซอร์โวมอเตอร์นั้น ทำได้โดยอาศัยความกว้างของพัลส์ที่ทำการป้อนให้เซอร์โวมอเตอร์ โดยสัญญาณพัลส์นี้จะเป็นสัญญาณ TTL โดยระดับสูง หรือ 1 จะมีแรงดัน 5VDC ระดับ ต่ำ หรือ 0 จะมีแรงดัน 0 VDC

**เซอร์โวมอเตอร์หมุนไปทางขวา** หรือ ตามเข็มนาฬิกา จะต้องสร้างสัญญาณพัลส์ ขนาด 1 ms เมื่อเซอร์โวมอเตอร์หมุนไปทางขวาสุดแล้วมอเตอร์จะหยุดหมุนเอง ถ้าป้อนพัลส์เข้าไปอีกจะเซอร์โวมอเตอร์จะไม่ทำงาน

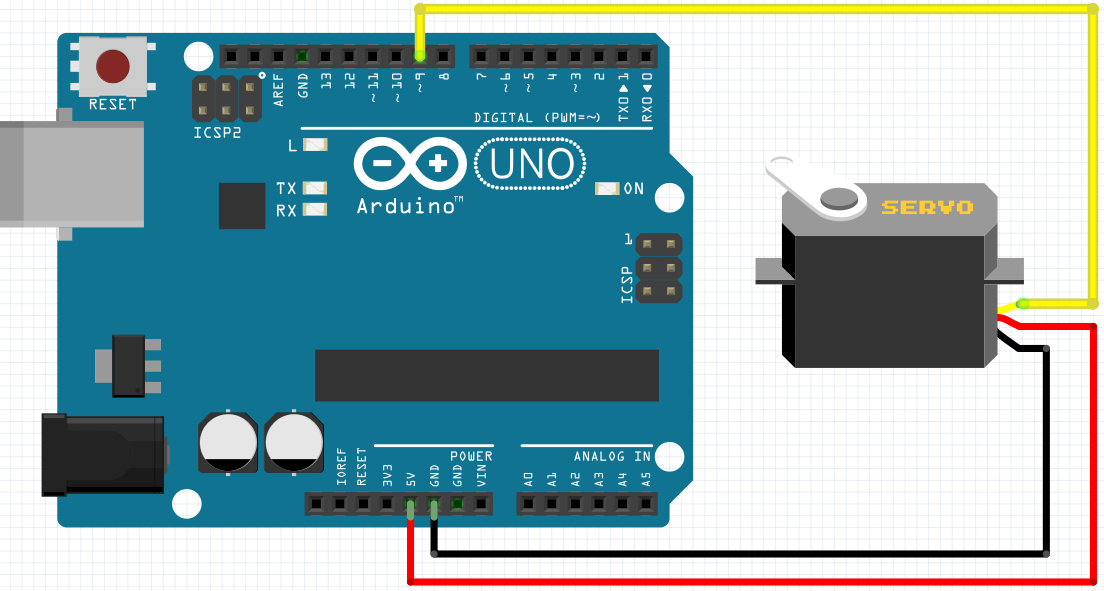
**เซอร์โวมอเตอร์หมุนไปทางซ้าย** หรือ ทวนเข็มนาฬิกา จะต้องสร้างสัญญาณพัลส์ ขนาด 2 ms เมื่อเซอร์โวมอเตอร์หมุนไปทางซ้ายสุดแล้วมอเตอร์จะหยุดหมุนเอง ถ้าป้อนพัลส์เข้าไปอีกจะเซอร์โวมอเตอร์จะไม่ทำงาน

**เซอร์โวมอเตอร์ไปตำหแหน่งกึ่งกล่าง** หากต้องการให้เซอร์โวมอเตอร์หนุนไปทางซ้าย หรือ ทวนเข็มนาฬิกา จะต้องสร้างสัญญาณพัลส์ ขนาด 1.5 ms

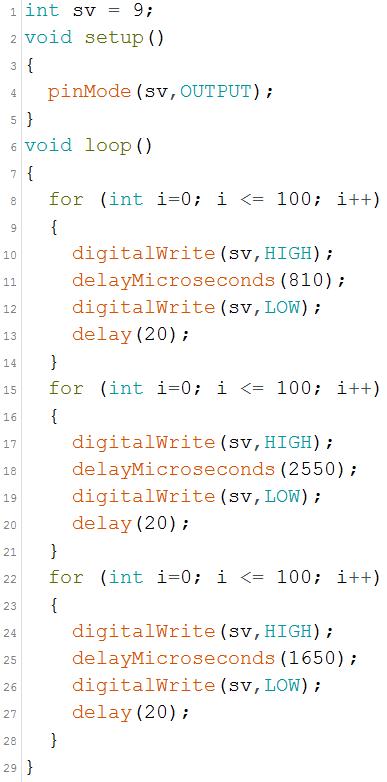


เซอร์โวมอเตอร์แต่ละตัวจะมีการทำงานของความกว้างของพัลส์ไม่เท่ากัน ค่าดังกล่าวเป็นค่าประมาณเท่านั้น ดังนั้นถ้าต้องการค่าที่ถูกต้อง ต้องดูที่คู่มือด้วยทุกครั้ง และจำนวนลูกสัญญาณพัลส์จะมีผลต่อองศาในการหมุนด้วยเช่นเดียวกัน

**ตัวอย่างการต่อวงจร และ Source Code การใช้งาน Servo Motor**



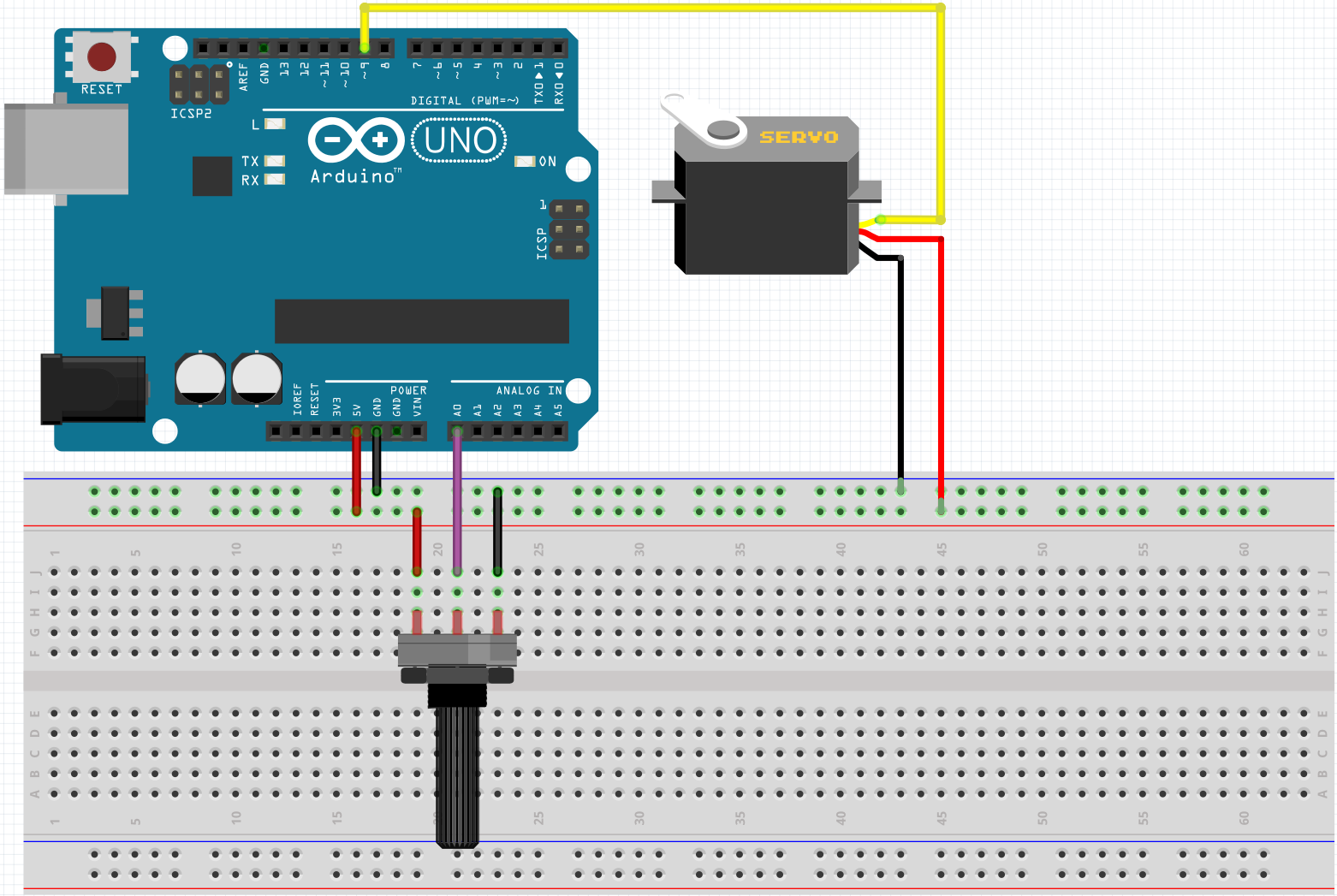
Source Code



**การทดลองที่ 1 จงเขียนโปรแกรมเพื่อควบคุมการหมุนของ Servo Motor จากตัวต้านทานปรับค่าได้**

ในการทดลองนี้ ใช้การรับค่าจากตัวต้านทางปรับค่าได้ VR และนำสัญญาณที่ได้มาควมคุมการหมุนของ Servo Motor โดยการหมุนตัวต้านทานปรับค่าได้ โดยค่าที่ได้คือ 0 = 0 องศา และ 1023 = 180 องศา เพื่อให้ Servo Motor หมุนตามองศาที่ปรับได้จากตัวต้านทานปรับค่าได้

รูปแบบการต่อวงจร



Source Code

int sv = 9;

int val;

void setup() {

pinMode(sv,OUTPUT);

}

void loop() {

val = map(analogRead(0),0,1023,810,2600);

for(int i =0;i <= 20;i++){

digitalWrite(sv,HIGH);

delayMicroseconds(val);

digitalWrite(sv,LOW);

delay(20);

}

}

บันทึกผลการทดลอง

หมุน องศา Servo ตามค่าที่อ่านได้จาก VR

**วิธีควบคุม Servo Motor ด้วยไลบรารี่ของ Arduino**

Arduino มีไลบรารี่สำหรับสั่งงาน RC Servo Motor มาให้ใช้งานอยู่แล้วเป็นฟังก์ชั่นสำเร็จรูปและใช้งานได้ง่าย ในหน้าเว็บไซต์ http://arduino.cc/en/reference/servo ได้ให้ข้อมูลไว้ว่า Servo Library ของ Arduino สามารถสั่งงาน Servo Motor ได้ทั้งแบบหมุนไป-กลับได้ 0-180 องศา และแบบต่อเนื่องที่หมุนครบรอบได้เรียกว่าเป็น Continuous Rotation Servo โดยสามารถรองรับการเชื่อมต่อ RC Servo Motor ได้ถึง 12 ตัวกับบอร์ด Arduino UNO และรองรับสูงสุดถึง 48 ตัวหากใช้บอร์ด Arduino Mega

ฟังก์ชั่นภายใน Servo Library

- attach()

- write()

- writeMicroseconds()

- read()

- attached()

- detach()

**attach()**

คือฟังก์ชั่นที่ใช้ในการกำหนดขาสัญญาณที่ Servo Motor ต่อกับ Arduino และกำหนดความกว้างของพัลซ์ที่ 0 องศาและ 180 องศา

Syntax

Servo.attach(pin)

Servo.attach(pin,min,max)

Parameters

Pin: คือ ขาสัญญาณของ Arduino ที่ใช้เชื่อมต่อกับ Servo Motor

Min: คือ ความกว้างของพัลซ์ที่ 0 องศาของ Servo ตัวที่ใช้ในหน่วยไมโครวินาที (us) โดยปกติแล้วหากไม่มีการตั้งค่าโปรแกรมจะกำหนดค่าไว้ที่ 544 us

Max: คือ ความกว้างของพัลซ์ที่ 180 องศาของ Servo ตัวที่ใช้ในหน่วยไมโครวินาที (us) โดยปกติแล้วหากไม่มีการตั้งค่าโปรแกรมจะกำหนดค่าไว้ที่ 2400 us

**Write()**

คือฟังก์ชั่นที่ใช้ควบคุมตำแหน่งที่ต้องการให้ Servo Motor หมุนไปยังองศาที่กำหนดสามารถกำหนดเป็นค่าองศาได้เลย คือ 0-180 องศา แต่ใน Servo Motor ที่เป็น Full Rotation คำสั่ง write จะเป็นการกำหนดความเร็วในการหมุน โดย

ค่าเท่ากับ 90 คือคำสั่งให้ Servo Motor หยุดหมุน

ค่าเท่ากับ 0 คือการหมุนด้วยความเร็วสูงสุดในทิศทางหนึ่ง

ค่าเท่ากับ 180 คือการหมุนด้วยความเร็วสูงสุดในทิศทางตรงกันข้าม

Syntax

servo.write(angle)

Parameters

Angle: คือมุมที่ต้องการให้ RC Servo Motor แบบ 0-180 องศาหมุนไป แต่หากเป็น RC Servo Motor แบบ Full Rotation ค่า Angle คือ การกำหนดความเร็วและทิศทางในการหมุน

**writeMicroseconds()**

คือฟังก์ชั่นที่ใช้ควบคุมตำแหน่งที่ให้ Servo Motor หมุนไปยังตำแหน่งองศาที่กำหนดโดยกำหนดเป็นค่าความกว้างของพัลซ์ในหน่วย us ซึ่งปกติแล้ว RC Servo Motor จะใช้ความกว้างของพัลซ์อยู่ที่ 1000-2000 us ตามที่ได้กล่าวไปข้างต้นแล้ว แต่ RC Servo Motor บางรุ่นหรือบางยี่ห้อไม่ได้ใช้ ช่วงความกว้างของพัลซ์ตามที่ได้กล่าวเอาไว้นี้ อาจจะใช้ช่วง 700-2300 แทนก็สามารถใช้ฟังก์ชั่น writeMicroseconds นี้เพื่อกำหนดความกว้างพัลซ์ได้เอง

*การใช้ฟังก์ชั่น writeMicroseconds สามารถกำหนดค่าได้อิสระ ตรงนี้ ”ต้องระวังในการใช้งาน” หากสั่งงาน RC Servo Motor (แบบ 0 - 180 องศา) จนหมุนไปเกินจุดสิ้นสุดคือเกินทั้งฝั่ง 0 หรือ 180 องศา จะทำให้เกิดเสียงครางดังจากการหมุนไปต่อไม่ได้และมอเตอร์จะกินกระแสสูงขึ้นด้วยในเวลาเดียวกันนั้น ซึ่งอาจทำให้ RC Servo Motor เกิดความเสียหายได้*

Syntax

servo.writeMicroseconds(uS)

Parameters

uS: คือค่าความกว้างของพัลซ์ที่ต้องการกำหนดในหน่วยไมโครวินาที (โดยตัวแปร int)

**read()**

คือฟังก์ชั่นอ่านค่าองศาที่สั่งเข้าไปด้วยฟังก์ชั่น write() เพื่อให้รู้ว่าตำแหน่งองศาสุดท้ายที่เราสั่งเข้าไปนั้นมีค่าเท่าไหร่ซึ่งค่าที่อ่านออกมานั้นจะมีค่าอยู่ในช่วง 0 - 180

Syntax

servo.read()

**attached()**

คือฟังก์ชั่นตรวจสอบว่า Servo ที่เราต้องการใช้กำลังต่ออยู่กับขสัญญาณของ Arduino หรือไม่

Syntax

servo.attached()

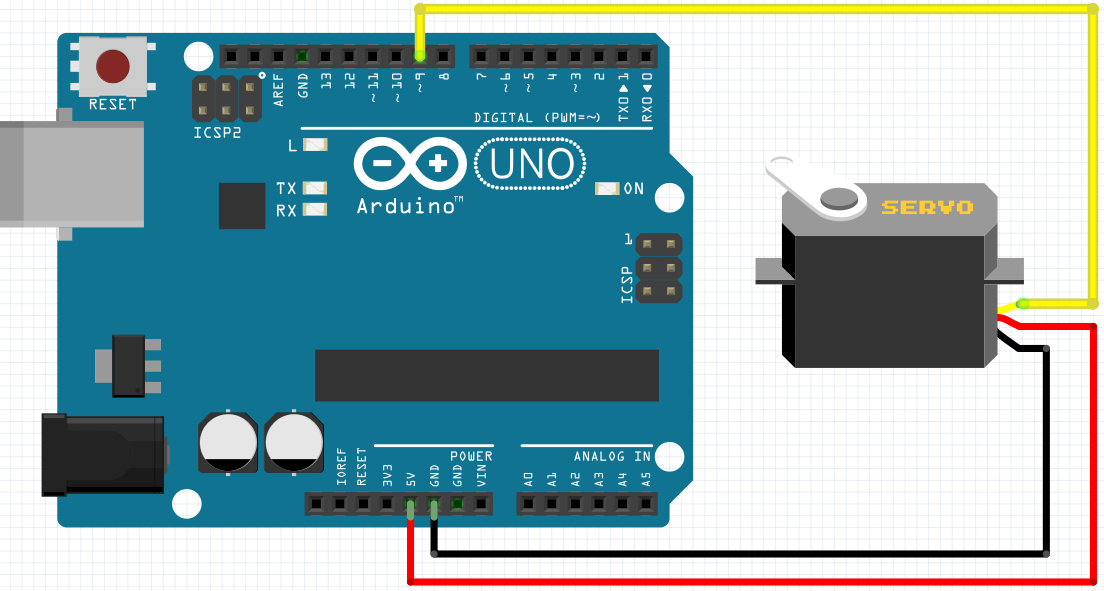
**detach()**

คือฟังก์ชั่นคืนสถานะของขาที่เรากำหนดให้เป็นขาควบคุม Servo Motor ด้วยคำสั่ง attached() ให้กลับคือสู่การใช่งานปกติ

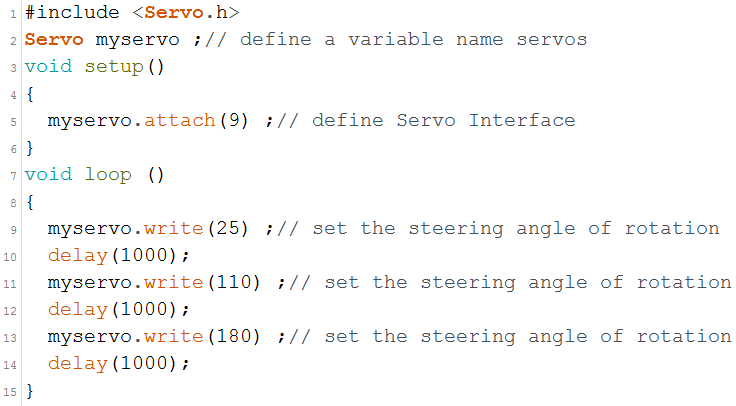
Syntax

servo.detach()

**ตัวอย่างการต่อวงจร และ Source Code การใช้งาน Servo Motor ด้วย Library <Servo.h>**



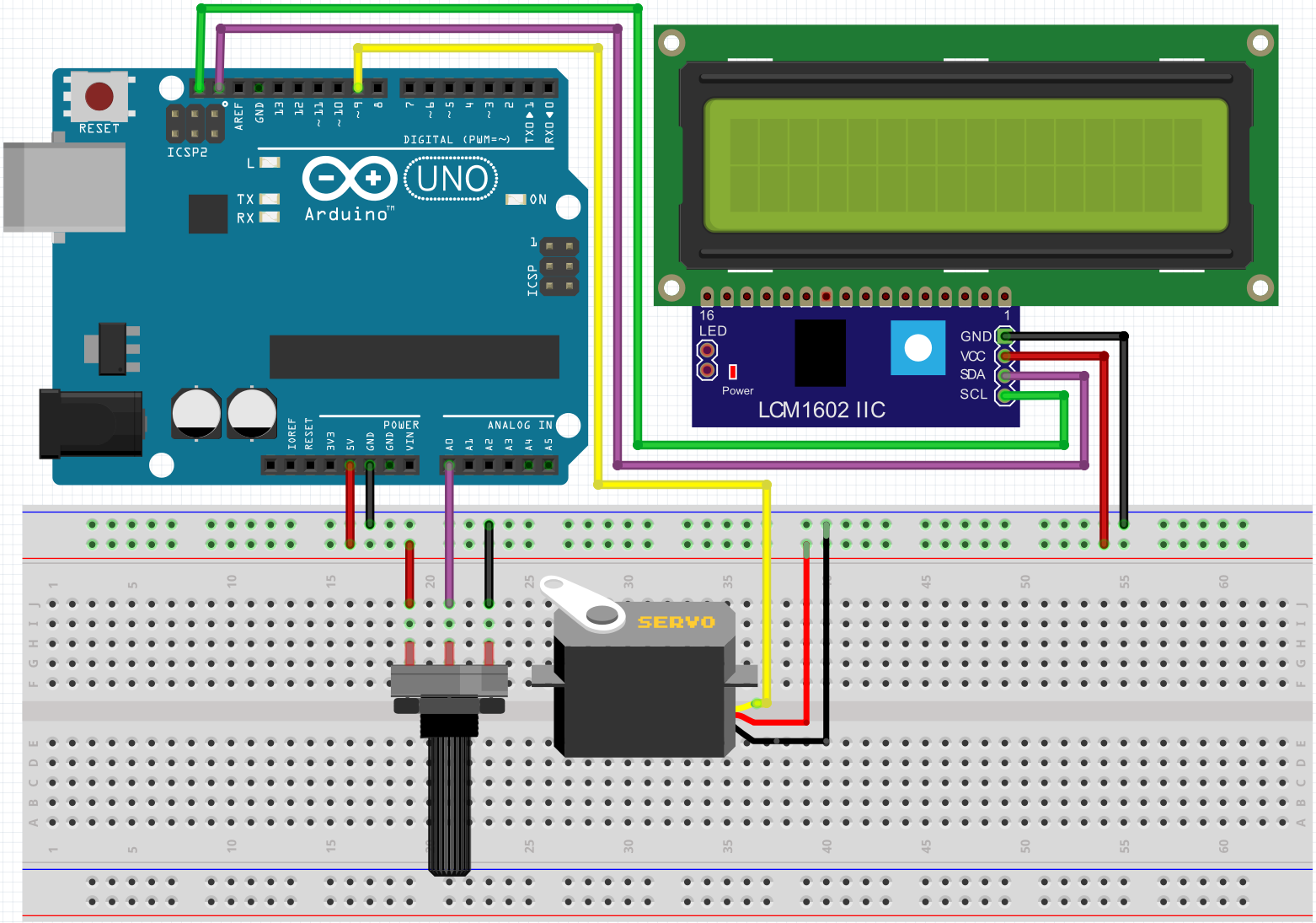
Source Code



**การทดลองที่ 2 จงเขียนโปรแกรมเพื่อควบคุมการหมุนของ Servo Motor จากตัวต้านทานปรับค่าได้ โดยใช้งาน Library <Servo.h>**

ในการทดลองนี้ ใช้การรับค่าจากตัวต้านทางปรับค่าได้ VR และนำสัญญาณที่ได้มาควมคุมการหมุนของ Servo Motor โดยการหมุนตัวต้านทานปรับค่าได้ โดยค่าที่ได้คือ 0 = 0 องศา และ 1023 = 180 องศา เพื่อให้ Servo Motor หมุนตามองศาที่ปรับได้จากตัวต้านทานปรับค่าได้ โดยใช้งาน Library <Servo.h> และแสดงผลองศาการหมุนของ Servo Motor บนหน้าจอ LCD

+รูปแบบการต่อวงจร



Source Code

#include <LiquidCrystal\_I2C.h>

#include <Wire.h>

#include <Servo.h>

Servo myServo;

int sv = 9;

int val,angel;

LiquidCrystal\_I2C lcd(0x27,16,2);

void setup() {

myServo.attach(sv,810,2600);

lcd.init();

lcd.backlight();

}

void loop() {

val = map(analogRead(0),0,1023,0,180);

myServo.write(val);

angel = myServo.read();

lcd.setCursor(0,0);

lcd.clear();

lcd.print("Angle : ");

lcd.setCursor(7,0);

lcd.print(angel);

delay(50);

}

บันทึกผลการทดลอง

หมุน องศา Servo ตามค่าที่อ่านได้จาก VR แล้วไปแสดงผลบน LCD