**Lab# 2**

การรับข้อมูลจากภายนอกเข้าสู่ ARDUINO

**การทดลองที่ 1** **การทดลองพอร์ตอินพุตดิจิตอลของ Arduino**

สำหรับขาพอร์ตที่กำหนดให้เป็นอินพุตผ่านทางฟังก์ชั่น pinMode() จะมีสถานะเป็น อิมพีแดนซ์สูงทำ ให้มีความต้องการกระแสไฟฟ้าจากอุปกรณ์ที่ต้องการอ่านค่าอินพุต น้อยมาก ทำให้ไม่สามารถรับหรือจ่ายกระแสให้กับวงจรภายนอกได้ แต่สามารถขาที่เป็นอินพุตนี้ไปใช้งานบางประเภท เช่น สร้างตัวตรวจจับการสัมผัส ที่อาศัยการวัด ค่าความจุไฟฟ้า

ก่อนที่จะใช้งานขาดิจิตอลของ Arduino จะต้องสั่งก่อนว่าให้ ขา Digital นี้ทำหน้าที่เป็นอินพุตหรือเอาต์พุต ในการทดลองนี้จะทดลองต่อเป็นอินพุต

**การกำหนดโหมดของขาพอร์ต**

ก่อนใช้งานต้องกำหนดโหมดการทำงานของขาพอร์ตดิจิตอลให้เป็นอินพุตหรือเอาต์พุต กำหนดได้จากฟังก์ชั่น pinMode() มีรูปแบบดังนี้

**pinmode(pin,mode);**

เมื่อ pin คือ หมายเลขขาที่ต้องการ

Mode คือ โหมดการทำงาน (INPUT หรื อ OUTPUT)

หลังจากที่กำหนดให้เป็นอินพุตแล้ว เมื่อต้องการอ่านค่าจากขานั้นๆ ให้เรียกใช้ฟังก์ชั่น digitalRead() โดยมี รูปแบบดังนี้

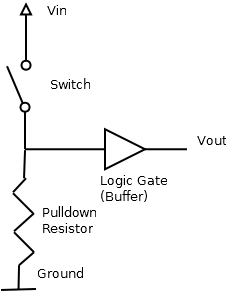
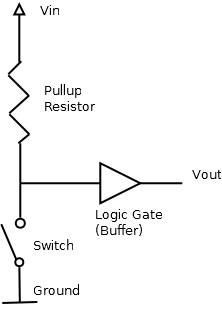
**digitalRead(pin);**

เมื่อ pin คือหมายเลขขาที่ ต้องการอ่านค่าสถานะ

เมื่อฟังก์ชั่นทำงานจะคืนค่าเป็นสถานะของขาที่ต้องการอ่านค่า โดยคืนค่าเป็น LOW (ค่าเป็น “0”) หรือ HIGH (ค่าเป็น “1”)

**การต่อตัวต้านทานแบบ pull-up และ pull-down**

สำหรับขาอินพุต เมื่อไม่มีอินพุตป้อนให้จะต้องกำหนดค่าแรงดันให้แน่นอน ทำได้โดยต่อตัวต้านทานพูลอัป (pull-up resistor) โดยต่อขาของตัวต้านทานขาหนึ่งไปยังไฟเลี้ยง หรือต่อพูลดาวน์ (pull- down) ซึ่งต่อขาหนึ่งของตัวต้านทานจากขาพอร์ตลงกราวด์ ค่าตัวต้านทานที่ ใช้ ทั่วไปคือ 10k



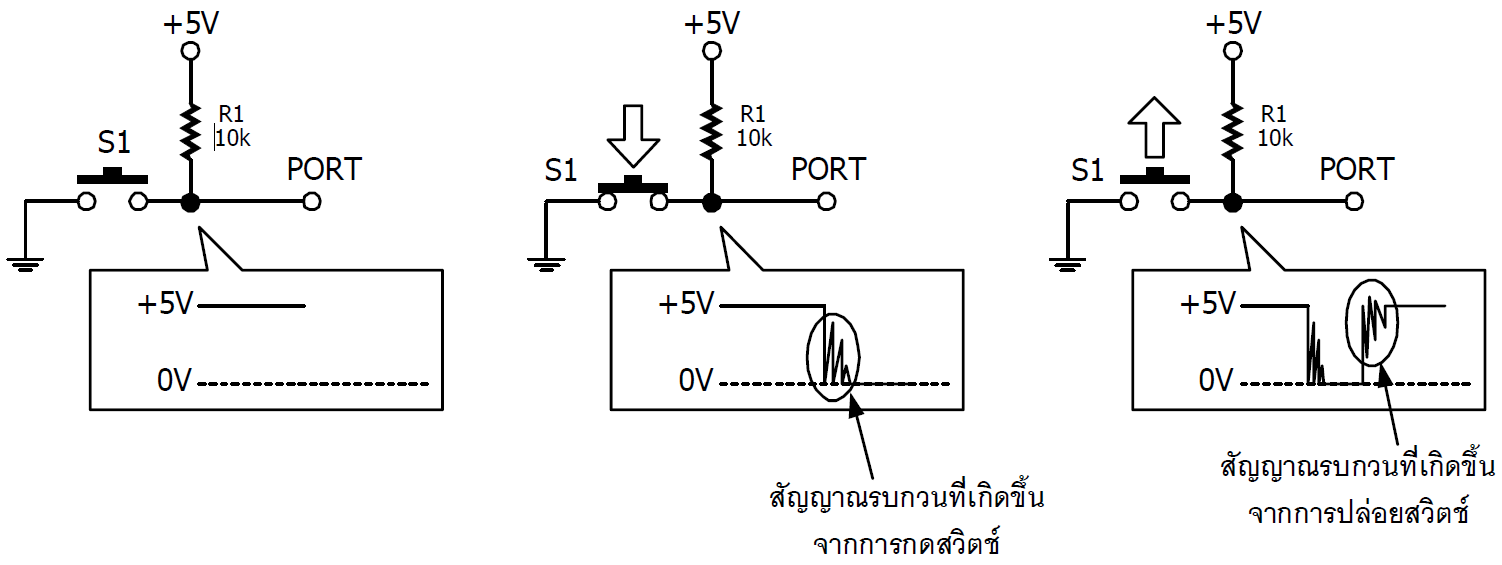
**รูปการต่อตัวต้านทานแบบ pull-up และ pull-down**

**Pull-up Resistor** คือการนำตัวต้านทานต่อเข้ากับ Vcc (+5V) เพื่อให้แรงดันอยู่คงที่ ทำให้อยู่ในสถานะ “HIGH” หรือ “1” ตลอดเวลา และเมื่อกดสวิตซ์ กระแสไฟฟ้าจะไหลลง Ground ทันที ซึ่งทำให้สถานะเป็นลอจิก “LOW” หรือ “0” และ การทำงานลักษณะนี้ จะเรียกว่า Active Low เพราะว่าจะเขียนโปรแกรมที่ทำงาน เมื่อลอจิกเป็น “LOW” ส่วนใหญ่ การต่อสวิตซ์ นิยมใช้แบบ Pull-up มากกว่า

**Pull-down Resistor** โดยใน Pull-down จะมีลักษณะคล้ายกับ Pull-up Resistor แตกต่าง ตรงที่ สภาวะปกติของ Pull-down จะเป็นลอจิก “LOW” หรือ “0” เมื่อมีการกดปุ่ม กระแสไฟจะไหลเข้าขาอินพุท ทำให้ ลอจิกเป็น “HIGH” หรือ “1” ได้ การทำงานในลักษณะนี้ จะเรียกว่า Active High

**Switch bounce**

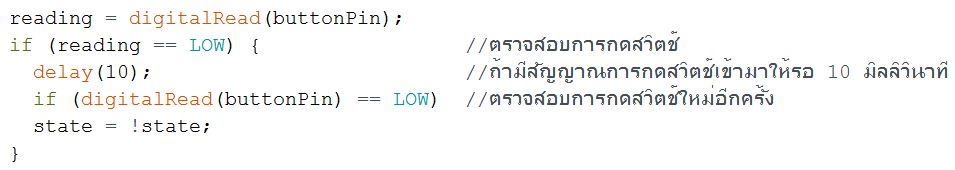
โดยทั่วไปแล้วสวิตช์ที่ใช้จะเป็นสวิตช์ทางกลที่ประกอบด้วยหน้าสัมผัสโลหะ มักจะมีปัญหาการ bounce ของสัญญาณในการกดสวิตช์ให้ต่อวงจร คืออาการไกว่ของสัญญาณ พบว่าหน้าสัมผัสของสวิตช์จะไม่สัมผัสกันสนิททันที โดยมีช่วงเวลาที่เริ่มสัมผัส และหลุดเป็นช่วงเวลาสั้นๆ 5-50 นาโนวินาที จะเกิดสัญญาณ bounce คืออาการที่สัญญาณจะสลับ เป็น High หรือ Low อย่างรวดเร็ว สังเกตจากกราฟ ก่อนที่หน้าสัมผัสของสวิตช์ จะต่ออย่างสมบูรณ์



**รูปแสดงการเกิดสัญญาณรบกวนเมื่อมีการกด และปล่อยสวิตช์ในวงจรดิจิตอล**

วิธีการแก้ไข คือ หลังจากหน้า contact หรือกดปุ่มในครั้งแรก arduino จะรับสัญญาณการกดปุ่มได้ แต่ยังไม่แน่ใจว่าสวิตซ์โดนกดจริง ให้รออีก 5-50 นาโนวินาที แล้วตรวจสอบสถานะการกดปุ่มอีกครั้ง ถ้ามีการกดจริง สัญญาณดิจิตอลที่อ่านได้ อยู่ในช่วงที่เสถียร ซึ่งค่าที่ออกมาจะบอกได้ว่ากด แต่ถ้าตรวจสอบอีกที่แล้วไม่พบ แสดงว่าสัญญาณที่ได้รับ เป็น noise ในระบบ

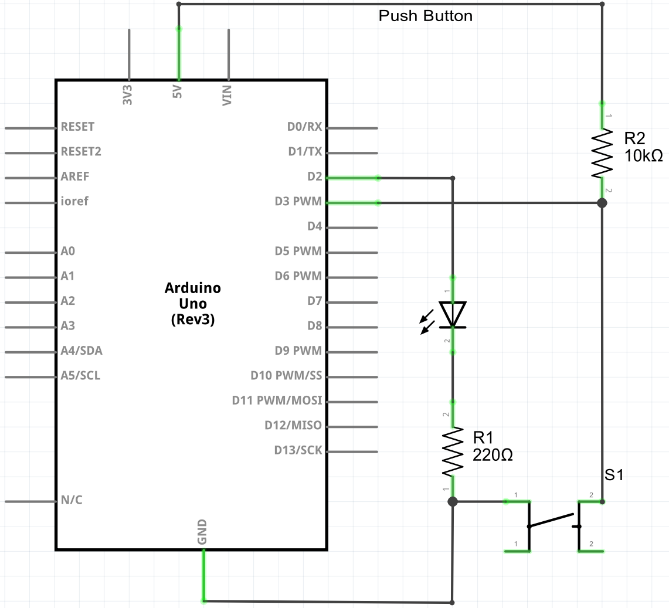
ตัวอย่าง Source Code

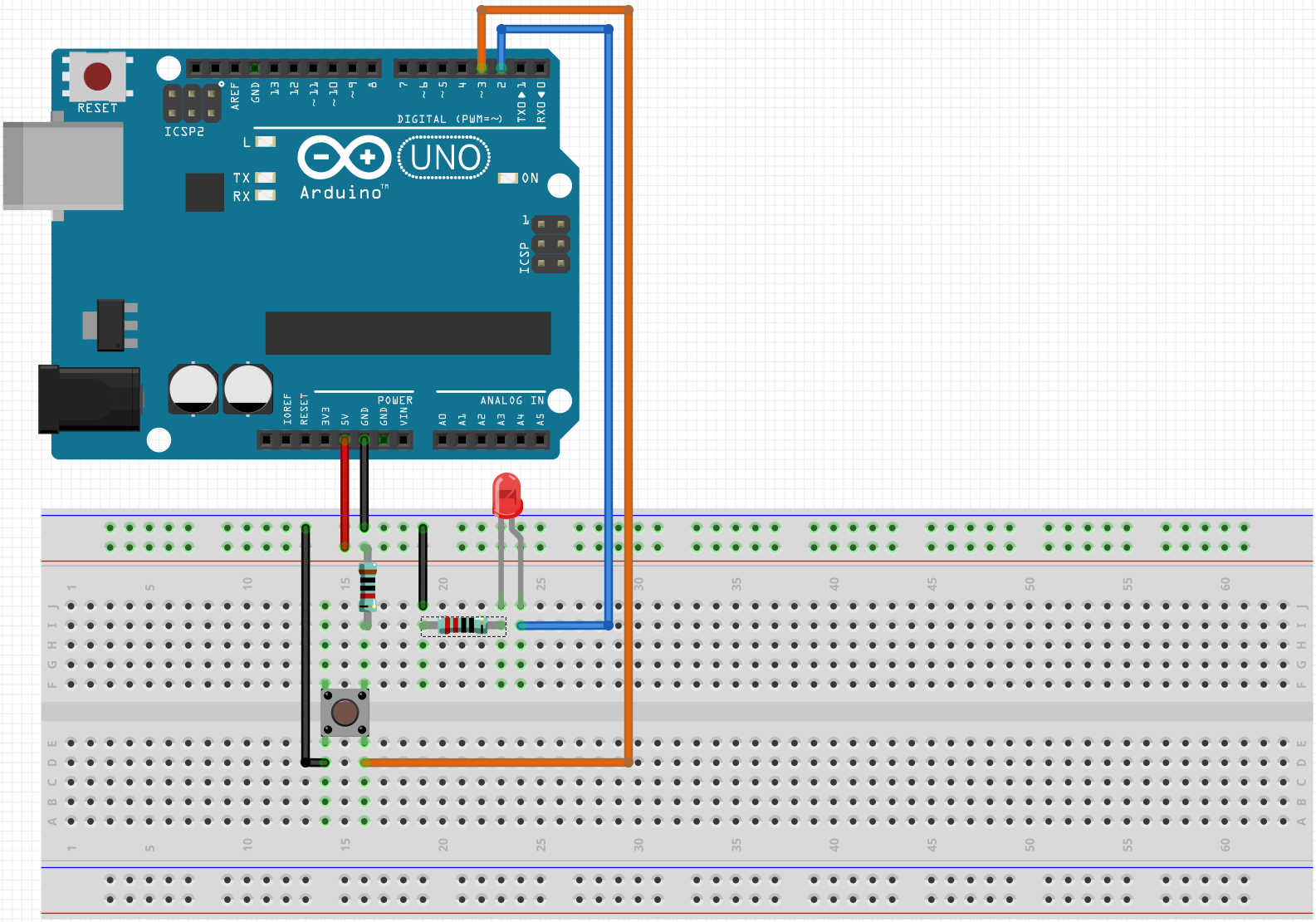


**1.1 ทดลองรับค่าจากสวิตช์**

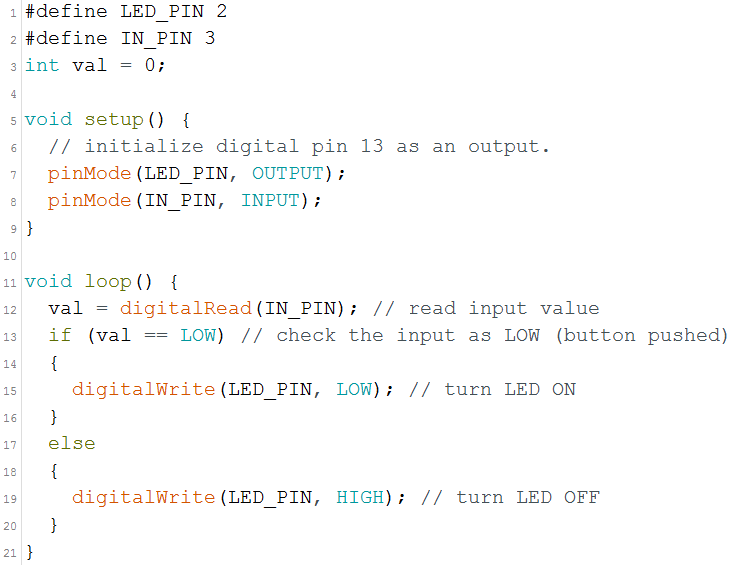
ในการทดลองนี้ จะทดลองอ่านค่าสถานะของสวิตช์แบบกดติดปล่อยดับ เพื่อควบคุมหลอด LED เมื่อ

กดสวิตช์ S1 ทำให้ LED1 ติดสว่าง เมื่อปล่อยสวิตช์ LED1 จะดับ

รูปแบบการต่อวงจร



Source Code

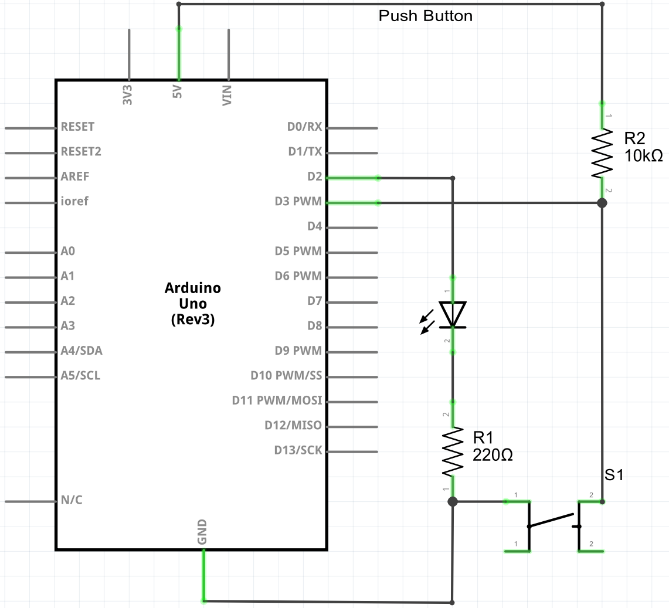


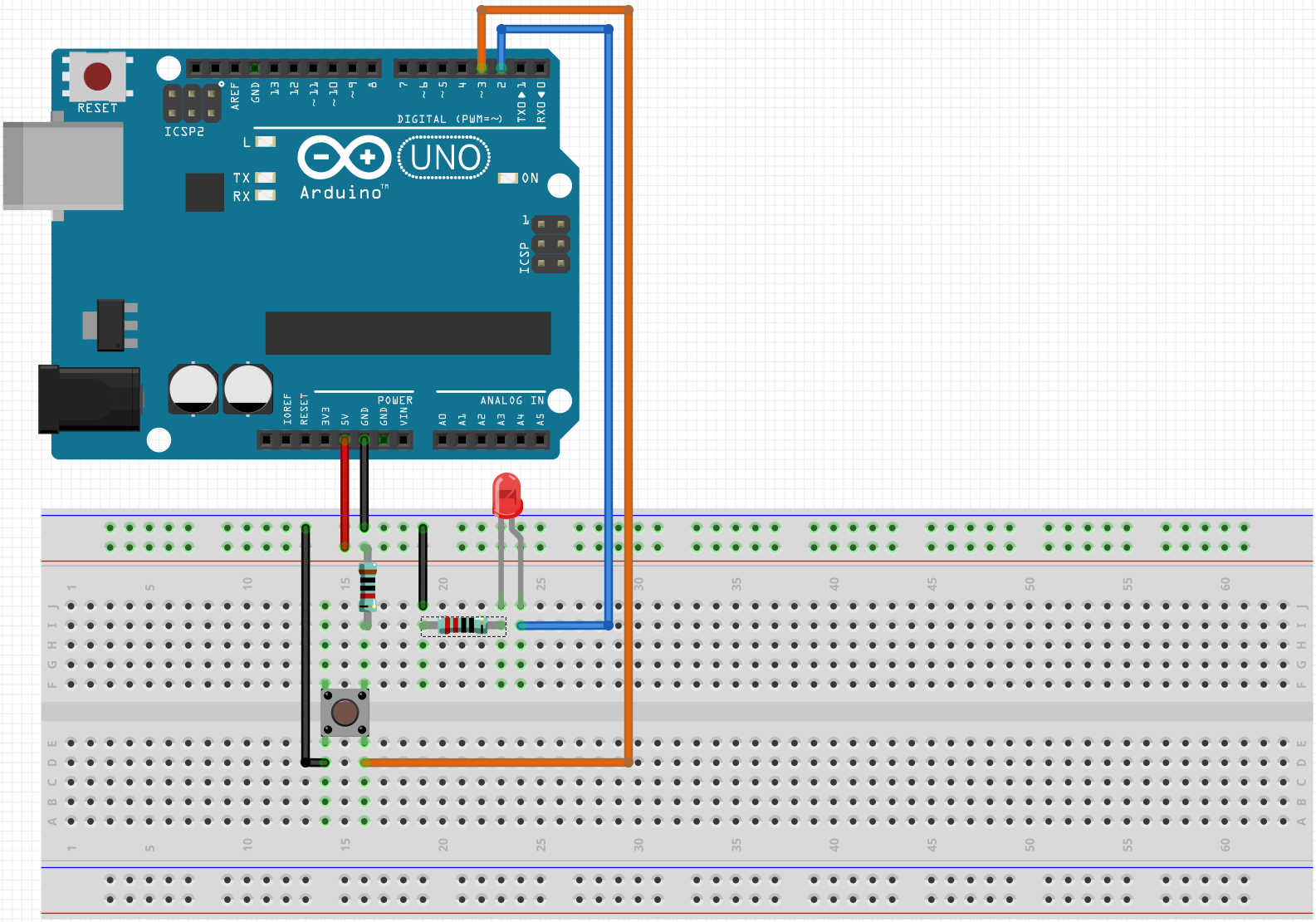
ทดลองแก้ไขโค้ดโปรแกรมบันทัดที่ 15 เป็น digitalWrite(LED\_PIN, HIGH) และแก้ไขบันทัดที่ 19 เป็น digitalWrite(LED\_PIN, LOW) ทดลองรัน และบันทึกผลการทดลอง

เริ่มแรกสถานะเป็น HIGH หลังจากกด switch กดค้างแล้วไฟดับ พอปล่อยแล้วไฟเปิด

**การทดลองที่ 2 จงเขียนโปรแกรมสวิตช์กดเปิดกดปิด (Toggle Switch)**

ในการทดลองนี้ ให้เขียนโปรแกรมเพื่อกดสวิตช์เพื่อเปิดปิดหลอด LED โดยที่เมื่อกดสวิตซ์หนึ่งครั้งหลอดไฟติดค้าง และกดอีกหนึ่งครั้งหลอดไฟจะดับ

รูปแบบการต่อวงจร



\*ตัวอักษรสีแดงคือ comment นะจ๊ะ

\*ตัวอักษรสีดำคือ Code

Source Code

#define LED\_PIN 2 // ขา 2 ขาที่ต่อเป็น OUTPUT ของหลอดไฟ

#define IN\_PIN 3 // ขา 3 จะใช้รับ INPUT จาก switch

int val = 0; // ตัวแปรที่ใช้เก็บค่าที่อ่านได้จาก switch

int temp = 0; // เก็บค่าสถานะของหลอดไฟ ณ ปัจจุบันค่าตั้งต้นคือ 0 คือหลอดไฟดับ

void setup() { // ฟังชั่น setup() ใช้ ตั้งค่าต่างๆ ฟังชั่นนี้ทำเพียงครั้งเดียวเท่านั้น

pinMode(LED\_PIN,OUTPUT); // ตั้งค่า pinMode ให้ขา LED\_PIN (ขา 2) มี Mode เป็น OUTPUT

pinMode(IN\_PIN,INPUT); // ตั้งค่า pinMode ให้ขา IN\_PIN (ขา 3) มี Mode เป็น INPUT

}

void loop() { // ฟังชั่น loop ทำซ้ำเรื่อยๆไม่มีเว้นวันหยุดราชการ และ วันหยุดนักขัตฤกษ์

val = digitalRead(IN\_PIN); // อ่านค่าจาก switch

if(val == LOW){ // ถ้าค่าที่อ่านได้เป็น LOW (เรากดปุ่มอยู่) จังหวะแรก

delay(50); // delay 50 millisecond รอสัญญาณกระเพื่อมเสร็จ

val = digitalRead(IN\_PIN); // อ่านค่าจาก switch อีกครั้ง การกด 1 ครั้งมี 2 จังหวะ จังหวะแรกคือ จังหวะที่เรากด ค่าจะเป็น LOW จังหวะที่ 2 คือจังหวะที่เราปล่อย ค่าจะเป็น HIGH

if(val == HIGH){ // ถ้าค่าที่เราอ่านได้เป็น HIGH (เราปล่อยแล้ว) จังหวะที่ 2

if(temp == LOW){ // ถ้าสถานะหลอดไฟที่เราเก็บไว้เป็น LOW คือ ปิดอยู่

digitalWrite(LED\_PIN,HIGH); //สั่งเปิดมันซะ

temp = 1; // เปลี่ยนค่าสถานะหลอดไฟ เปิด

}else{ // ถ้าสถานะหลอดไฟที่เราเก็บไว้เป็น HIGH คือเปิดอยู่

digitalWrite(LED\_PIN,LOW); // สั่งปิดมันซะ

temp = 0; //เปลี่ยนค่าสถานะเป็น ปิด

}

}

}

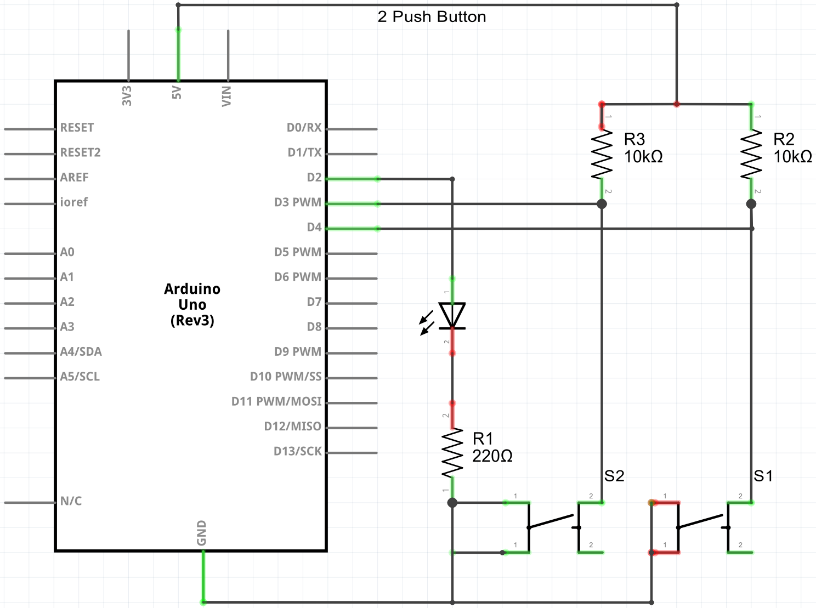
}

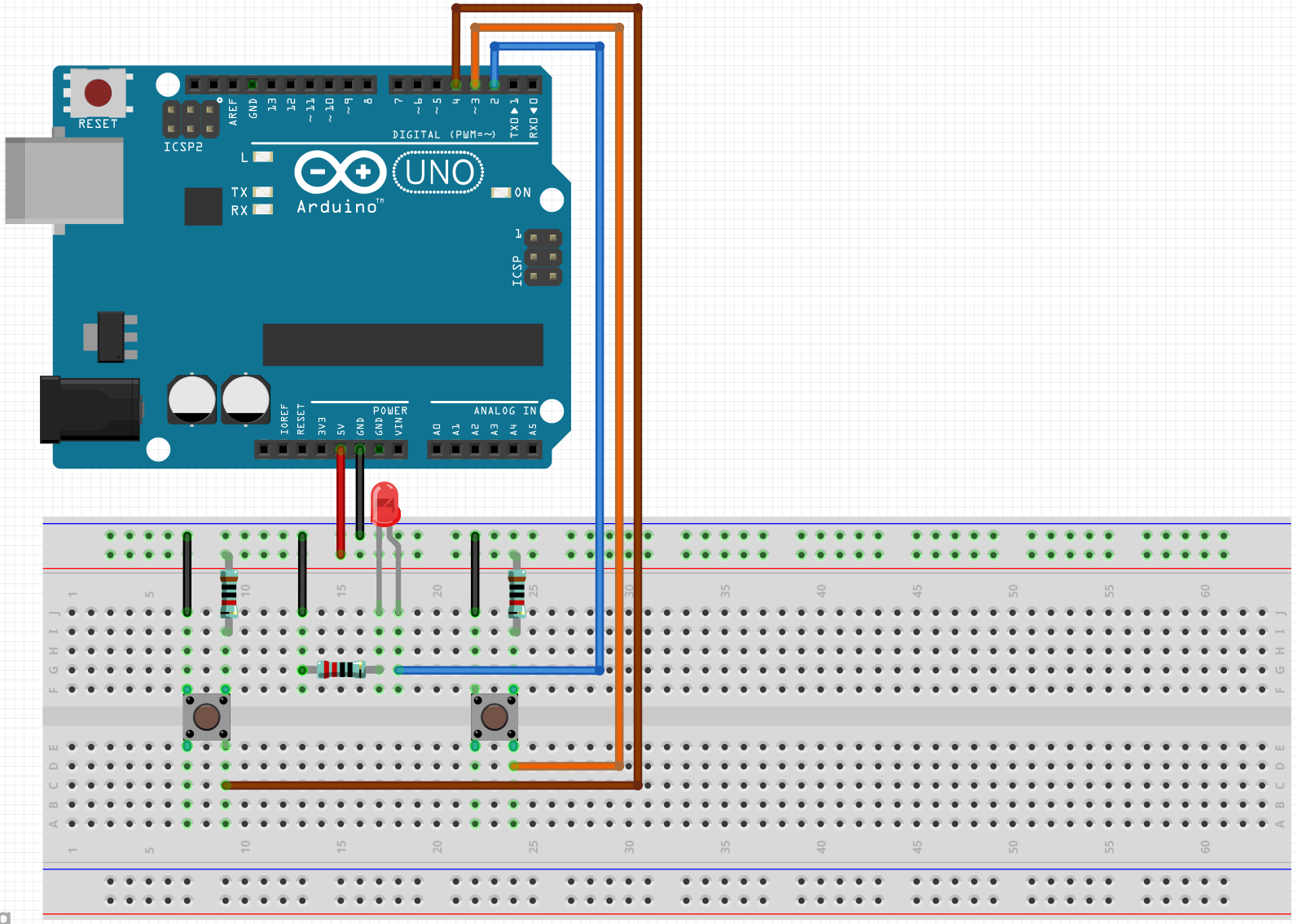
บันทึกผลการทดลอง

เริ่มแรกไฟปิด พอกด 1 ครั้ง \*\*การกดใน 1 ครั้ง มี 2 จังหวะคือ กด และ ปล่อย เช็คเงื่อนไขการกดครั้งแรกว่า ปิดก่อน แล้ว รอสัญญาณ จะหยุดสั่น แล้วค่อยเช็คว่ากดปล่อย แล้ว ค่อยเช็คว่า ไฟเปิดอยู่หรือเปล่า ถ้าเปิดอยู่ก็สั่งปิด ถ้าปิดอยู่ก็สั่งเปิด

**การทดลองที่ 3 จงเขียนโปรแกรมเพื่อสั่งเปิดปิดหลอด LED ด้วยสวิตช์ 2 ตัว**

ในการทดลองนี้ ให้เขียนโปรแกรมเพื่อกดสวิตช์ S1 เพื่อเปิดหลอด LED และกดสวิตช์ S2 เพื่อปิดหลอด LED

รูปแบบการต่อวงจร



Source Code

#define LED\_PIN 2 // ขา 2 ขาที่ต่อเป็น OUTPUT ของหลอดไฟ

#define IN\_PIN 3 // ขา 3 จะใช้รับ INPUT จาก switch 1

#define IN\_PIN2 4 // ขา 4 จะใช้รับ INPUT จาก switch 2

int val = 0; //เป็นตัวแปรที่เก็บค่าที่อ่านได้จาก switch 1

int val2 = 0; //เป็นตัวแปรที่เก็บค่าที่อ่านได้จาก switch 2

int temp = 0; //ตัวแปรเก็บค่าสถานะของหลอดไฟ

void setup() {

pinMode(LED\_PIN,OUTPUT); // ตั้งค่า ขา 2 เป็น OUTPUT

pinMode(IN\_PIN,INPUT); //ตั้งค่า ขา 3 เป็น INPUT

pinMode(IN\_PIN2,INPUT); //ตั้งค่าขา 4 เป็น INPUT

}

void loop() {

val = digitalRead(IN\_PIN); // อ่านค่าจาก switch 1

if(val == LOW){ // check การกด จังหวะที่ 1 (กด)

delay(50); // delay 50 milisecond เพื่อรอสัญญาณกระเพื่อมเสร็จ

val = digitalRead(IN\_PIN); // อ่านค่าจาก switch 1 อีกครั้ง

if(val == HIGH){ // check การกดจังหวะที่ 2 (ปล่อย)

if(temp == LOW){ //check ถ้าหลดไฟปิดอยู่

digitalWrite(LED\_PIN,HIGH); //ก็สั่งเปิดมันซะ

temp = 1; // เปลี่ยนค่าสถานะหลอดไฟเป็น เปิด

}

}

}

val2 = digitalRead(IN\_PIN2); // อ่านค่าจาก switch 2

if(val2 == LOW){ // check การกด จังหวะที่ 1 (กด)

delay(50); // delay 50 milisecond เพื่อรอสัญญาณกระเพื่อมเสร็จ

val2 = digitalRead(IN\_PIN2); // อ่านค่าจาก switch 2 อีกครั้ง

if(val2 == HIGH){ // check การกดจังหวะที่ 2 (ปล่อย)

if(temp == HIGH){ //check ถ้าหลอดไฟเปิดอยู่

digitalWrite(LED\_PIN,LOW); // ก็ปิดมันดิรอไร

temp = 0; //เปลี่ยนสถานะเป็นปิด ให้มันด้วย

}

}

}

}

บันทึกผลการทดลอง

ปุ่ม S1 เปิดไฟแต่ไม่สามารถปิดไฟได้ แต่จะปิดได้ก็ต่อเมื่อ กดปุ่ม S2

\*\*for คนบนฟ้า

ฉันไม่เคยดัง และยังคงเดินตอนนี้ หรือว่าตอนไหน  
ฉันก็คนหนึ่ง แต่รักฉันยิ่งใหญ่ ใหญ่กว่าใคร เธอคงรู้ดี

ยังคงมีแค่เธอและต่อไป ยังจะมีแค่เธอจนสุดท้าย  
กาย แตกสลาย จนวันที่ฉันหมดลม จากไป  
เชื่อเถอะความรักไม่เคยห่างหาย