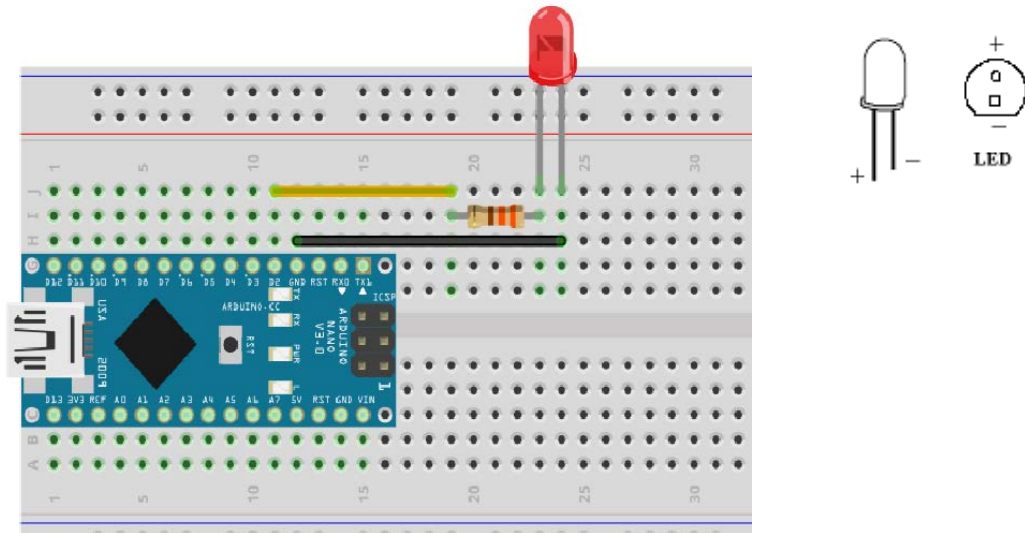


การใช้คำสั่ง digitalWrite กับขาที่เลือกโหมดเป็น input จะเป็นการเปิด-ปิด การต่อ pull-up ภายในวงจรของไมโครคอนโทรลเลอร์ โดย HIGH เป็นการเปิดโหมดการต่อ Pull-up และ LOW เป็นการปิดโหมด pull-up

ในกรณีการใช้คำสั่ง digitalWrite เพื่อสั่งให้ LED สว่าง โดยที่ไม่ได้กำหนดโหมดการทำงานของขาด้วยคำสั่ง pinMode จะส่งผลให้ LED ที่ต่ออยู่กับขานั้นไม่สว่างเท่าที่ควร เพราะการไม่ใช้คำสั่ง pinMode จะเปิดการทำงานโหมด pull-up ซึ่งจะทำให้กระแสไฟบางส่วนไหลผ่านตัวต้านทานภายในโดยไม่ผ่าน LED



รูปการเชื่อมต่ออุปกรณ์การทดลองลงบนโปรโตบอร์ด

1. ให้เชื่อมต่อสาย USB ของบอร์ดกับคอมพิวเตอร์ เปิดโปรแกรม Arduino ที่ได้ติดตั้งในคอมพิวเตอร์ จากนั้นทำการเขียนโปรแกรมที่ทำหน้าที่สั่งงานให้ LED ที่อยู่บนบอร์ดไมโครคอนโทรลเลอร์กะพริบทุก 1 วินาที จากนั้นทำการคอมไพล์แล้วทำการ Upload โปรแกรมที่ได้ลงบนบอร์ด Arduino

```
int led = 13;                                     // LED connected to digital pin 13

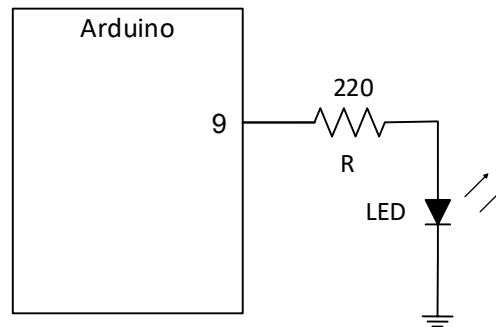
void setup()
{
  pinMode(led, OUTPUT);                           // initialize the digital pin as an output
}

void loop()
{
  digitalWrite(led, HIGH);                         // turn the LED on (HIGH is the voltage level)
  delay(1000);                                     // wait for a second (1000 milliseconds)
  digitalWrite(led, LOW);                          // turn the LED off by making the voltage LOW
  delay(1000);                                     // wait for a second (1000 milliseconds)
}
```

คำสั่งที่ใช้มีความหมายดังนี้

int led = 13;	ใช้ในการกำหนดขาที่ต่อ LED ภายในบอร์ด Arduino ว่าต่ออยู่ที่ขา 13
pinMode(led, OUTPUT);	กำหนดให้ขาที่ต่อ LED เป็นขาเอาต์พุต
digitalWrite(led, HIGH);	ให้ส่งค่าลอจิก 1 ออกไปขาที่ต่อกับ LED
digitalWrite(led, LOW);	ให้ส่งค่าลอจิก 0 ออกไปขาที่ต่อกับ LED
delay(1000);	ให้ทำการหน่วงเวลา 1000 ms (Milliseconds)

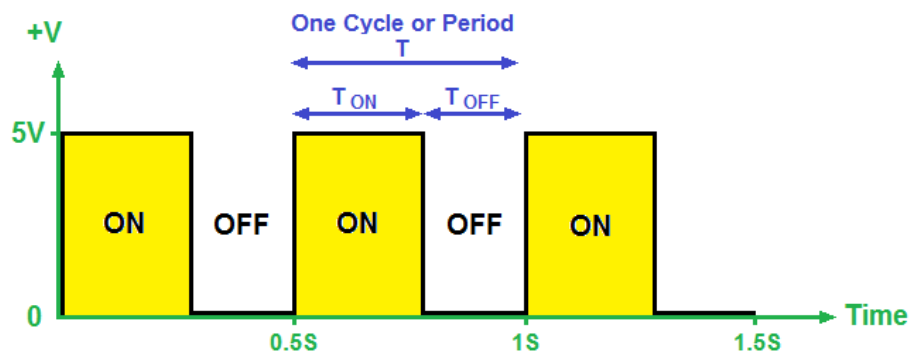
2. ให้ทำการย้ายขา LED ของโปรแกรมที่ ต่ออยู่ที่ 13 ไปเป็นขาที่ 9 และให้ต่อ LED อนุกรมกับตัวความต้านทาน 220  $\Omega$  เข้ากับขาที่ 9 แล้วลงกราวด์



3. จากข้อ 1 ให้แก้ไขโปรแกรมให้ LED กระพริบเป็นความถี่ 10 Hz

โดยที่ความถี่ (Frequency) เป็นจำนวนรอบที่แสดงว่าคลื่นเคลื่อนที่ไปได้กี่รอบในหนึ่งวินาที (Second) มีหน่วยเป็น รอบต่อวินาทีหรือเฮิรตซ์ (Hz) ใช้แทนสัญลักษณ์ด้วย  $f$

คาบเวลา (Period) คือ เวลาที่ใช้ในการเคลื่อนที่ครบ 1 รอบ (One Cycle) มีหน่วยเป็นวินาที (Second) ใช้แทนสัญลักษณ์ด้วย  $T$



จากรูปเมื่อเวลาผ่านไปหนึ่งวินาที คลื่นเคลื่อนที่ได้สองลูก แสดงว่าคลื่นนี้มีค่าความถี่ 2 Hz

หรือถ้าพิจารณาจากคาบเวลาจะเห็นว่าใน 1 รอบจะใช้เวลา  $T = T_{ON} + T_{OFF} = 0.5 \text{ Sec}$

ดังนั้นจะได้ความสัมพันธ์ระหว่างความถี่ ( $f$ ) และคาบ ( $T$ ) ตามสมการ

$$f = \frac{1}{T}$$

$$f = \frac{1}{0.5}$$

$$f = 2 \text{ Hz}$$

$$f = \frac{1}{T}$$

$$10 = \frac{1}{T}$$

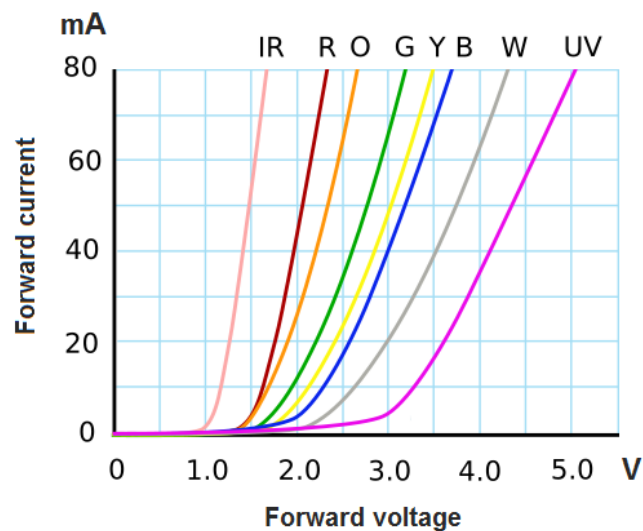
$$T = \frac{1}{10} = 0.1 \text{ s} = 0.1 \times 10^3 \text{ ms} = 100 \text{ ms}$$

$$\begin{aligned} T &= T_{on} + T_{off} \\ 100 &= T_{on} + T_{off} \\ \therefore T_{on}, T_{off} &= 50 \text{ ms} \end{aligned}$$

4. ให้วัดแรงดันไฟฟ้าตกคร่อมตัว LED ในช่วงที่ LED กำลังทำงาน (LED ON) โดยที่ LED จะต้องต่ออนุกรมกับตัวความต้านทาน 220  $\Omega$

LED	Forward Voltage
infrared (IR)	1.19 V
Red	2.03 V
Green	2.05 V
Blue	2.82 V
Ultraviolet (UV)	3.09 V

ค่าแรงดันไฟฟ้าที่ตกคร่อม LED ขณะป้อนแรงดันไฟฟ้าแบบตรงตามขั้ว (Forward) จะแปรผันตามกระแสที่ไหลผ่าน และจะขึ้นอยู่กับค่าความยาวคลื่นของแสงที่ส่องสว่างออกมาจาก LED ด้วย ดังตัวอย่างจะเป็นกราฟแสดงค่า Characteristic ของ LED แต่ละสี



5. ให้คำนวณหากระแสที่ไหลผ่าน LED ที่ใช้ทดลองมา 2 ตัวในช่วงขณะที่ LED กำลังทำงาน (LED ON) โดยใช้กฎของโอห์ม

$V = IR$ $I = \frac{V}{R}$ $I = \frac{5 - V_{LED}}{220}$ $I = \frac{5 - 2.03}{220}$ $I = 0.0135 \text{ A}$ $(Red)$	$V = IR$ $I = \frac{V}{R}$ $I = \frac{5 - V_{LED}}{R}$ $I = \frac{5 - 2.82}{220}$ $I = 0.01 \text{ A}$ $(Blue)$
--	---

กฎของโอห์ม (Ohm's Law) กล่าวว่าถ้ากระแสไฟฟ้าที่ไหลในตัวนำไฟฟ้าจะแปรผันตามแรงดันที่ตกคร่อมตัวนำนั้น และจะแปรผกผันกับค่าความต้านทานของตัวนำนั้น ดังสมการ

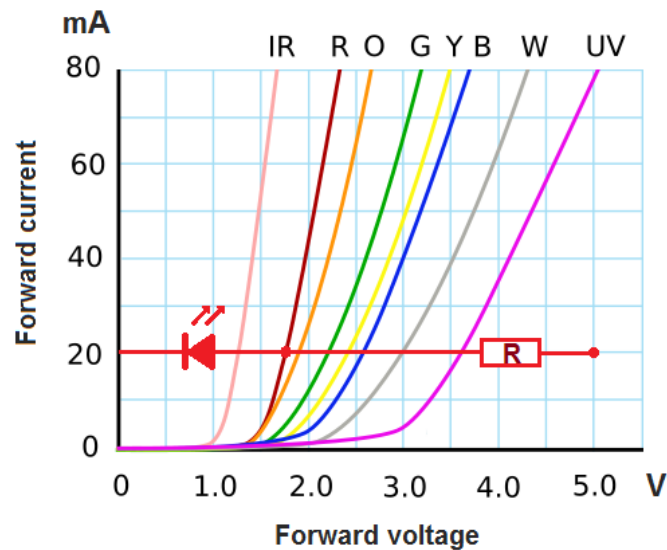
$$I = V / R$$

เมื่อ  $I$  = กระแสไฟฟ้ามีหน่วยเป็นแอมป์แปร์ (A)

$V$  = แรงดันไฟฟ้ามีหน่วยเป็นโวลต์ (V)

$R$  = ความต้านทานมีหน่วยเป็นโอห์ม ( $\Omega$ )

โดยที่แรงดันตกคร่อมตัวความต้านทานได้จากแรงดันของแหล่งจ่ายไฟลบด้วยแรงดันตกคร่อม LED มีความสัมพันธ์ดังรูป



6. ให้ทดลองทำการเปลี่ยนค่าความต้านทานจาก 220  $\Omega$  ไปเป็น 1K  $\Omega$  แล้วให้อธิบายผลที่ได้เป็นอย่างไร

แสงสว่างของหลอด LED ลดลงเนื่องจากกระแสไหลผ่านหลอด LED ลดลง

7. ให้แสดงวิธีการคำนวณหาความต้านทานที่เหมาะสม เมื่อกำหนดให้ Forward Current ของ LED เท่ากับ 20 mA

$$V = IR \quad 2.97 = 20 \times 10^{-3} R \quad V_{LED} = 2.03 V \text{ (Red)}$$

$$V_{in} - V_{LED} = IR \quad R = 148.5 \Omega$$

$$5 - 2.03 = (20 \times 10^{-3}) R$$

8. ให้ทำการแก้ไขโปรแกรมโดยการเปลี่ยนค่า delay() เพื่อให้ LED ติด สว่าง 0.5 วินาที และดับ 1.5 วินาที จากนั้นให้ LED กระพริบเร็วขึ้นเรื่อยๆ ตามลำดับจนกว่าเราจะไม่เห็นการกระพริบ โดยใช้คำสั่ง for (....)

9. จากข้อ 8 ค่าความถี่ในขณะที่เราเริ่มจะไม่เห็น LED กระพริบคือความถี่เท่าไร

$$T_{on} + T_{off} = T = 12 ms \quad \therefore f = \frac{1}{T} = \frac{1}{12 \times 10^{-3}} = \frac{1000}{12} = 83.3 \text{ Hz}$$

10. จากข้อ 8 ค่าความสว่างของ LED ในขณะที่เราเริ่มจะไม่เห็น LED กระพริบ ความสว่างนั้นเท่าเดิมหรือน้อยลง และให้เหตุผลว่าทำไมจึงเป็นเช่นนั้น

```

1 //9
2 int led = 13;
3
4 void setup()
5 {
6   pinMode(led, OUTPUT);
7 }
8
9 void loop()
10 {
11   for (int i = 0; i < 500; i++)
12   {
13     digitalWrite(led, HIGH);
14     delay(500 - i); //อนึ่ง
15     digitalWrite(led, LOW);
16     delay(1500 - i*3); //อนึ่ง
17   }
18   //Ton + Toff = 3 + 9 = 12ms
19   //500 i = 12ms เป็นค่าที่น้อยลงเรื่อยๆจนกว่าเราจะมองไม่เห็น จึงสรุปได้ว่าความถี่ + เวลา คือ 12ms
20   //จากนั้นก็วนกลับไปต้นสุดของ loop
21 }

```

ผู้เขียน แล้วยังให้เหตุผลว่าแสงสว่างจะเปลี่ยนไป

ความสว่างเท่าเดิม เนื่องจากใช้หลอดไฟจากแหล่งกำเนิดเท่าเดิม

11. ให้ต่อ LED หลอดที่ 2 อนุกรมกับตัวความต้านทาน 220  $\Omega$  เข้ากับขาที่ 10 แล้วลงกราวด์  

$$T_r = \frac{1}{f} = \frac{1}{1} s$$

$$T_o = \frac{1}{f} = \frac{1}{2} s$$
12. ให้เขียนโปรแกรมให้ LED ขาที่ 9 กระพริบเป็นความถี่ 1 Hz และให้ LED ขาที่ 10 กระพริบเป็นความถี่ 2 Hz
13. ให้ต่อ LED อนุกรมกับตัวความต้านทาน 220  $\Omega$  เพิ่มอีกเป็นจำนวน 5 หลอด แล้วให้เขียนโปรแกรมควบคุมให้หลอดไฟ LED กระพริบไล่จากขวาไปซ้าย แล้วกระพริบไล่จากซ้ายสุดและขวาสุดกลับกันไปมา โดยใช้คำสั่ง for (....)

การสื่อสารกับอุปกรณ์ภายนอกของบอร์ด Arduino จะใช้พอร์ตที่เรียกว่าพอร์ตอนุกรม (Serial Port) ในการเชื่อมต่อกับอุปกรณ์อื่น หรือสื่อสารระหว่างไมโครคอนโทรลเลอร์กับคอมพิวเตอร์ การสื่อสารนี้เรียกว่า UART โดยจะใช้ขาหมายเลข 0 (RX) ในการรับค่า และขาหมายเลข 1 (TX) ในการส่งค่า คำสั่งต่างๆที่จำเป็นมีดังนี้

void serial.begin(rate) เป็นการกำหนดอัตราของการรับส่งข้อมูล หน่วยเป็น bits per second (baud rate)  
 int serial.available() ใช้ตรวจสอบว่าบัฟเฟอร์รับข้อมูลไว้จำนวนกี่ไบต์  
 int serial.read() อ่านค่าข้อมูลที่ถูกส่งเข้ามายังพอร์ตอนุกรมของไมโครคอนโทรลเลอร์  
 void Serial.flush() เคลียร์บัฟเฟอร์ของพอร์ตอนุกรมให้ว่าง  
 void Serial.print() พิมพ์ข้อมูล ออกทางพอร์ตอนุกรม  
 void Serial.println() พิมพ์ข้อมูล ออกทางพอร์ตอนุกรมและขึ้นบรรทัดใหม่

14. ให้เพิ่มคำสั่ง Serial.begin(115200); // initialize serial communication at 115200 bits per second  
 ลงใน void setup() เพื่อใช้กำหนดอัตราความเร็วในการรับส่งข้อมูลผ่าน Serial Monitor มีค่าเท่ากับ 115200 bps
15. ให้เพิ่มคำสั่ง int Temp = analogRead(A0); // read the input on analog pin 0  
 ลงใน void loop() เพื่อใช้รับค่าสัญญาณอนาล็อกจากขา A0 ของบอร์ด Arduino และแปลงค่าที่ได้ไปเป็นสัญญาณดิจิทัลขนาด 10 บิต แล้วเก็บไว้ในตัวแปร Temp ซึ่งค่าที่ได้จะอยู่ระหว่าง 0 ถึง 1023 (คำนวณได้จาก  $2^{10}$ )
16. ให้เพิ่มคำสั่ง Serial.println(Temp); // print out the value  
 ต่อจากคำสั่งในข้อ 15 เพื่อให้พิมพ์ผลลัพธ์ค่าข้อมูลตัวแปร Temp ส่งออกไปทาง Serial Monitor