

## หน่วยที่ 8: การโปรแกรมใช้งานโมดูลอัลตราโซนิก

### จุดประสงค์

- สามารถเขียนโปรแกรมใช้งานโมดูลอัลตราโซนิกได้
- สามารถควบคุมการทำงานและแสดงผลผ่าน Blynk ได้

## กิจกรรม

## 8.1 กิจกรรม 1: การอ่านค่าจากโมดูลอัลตราโซนิกผ่าน Blynk

### โจทย์กิจกรรม

ให้ผู้เรียนเขียนโปรแกรมอ่านค่าจากโมดูลอัลตราโซนิกและแสดงผลผ่าน Blynk

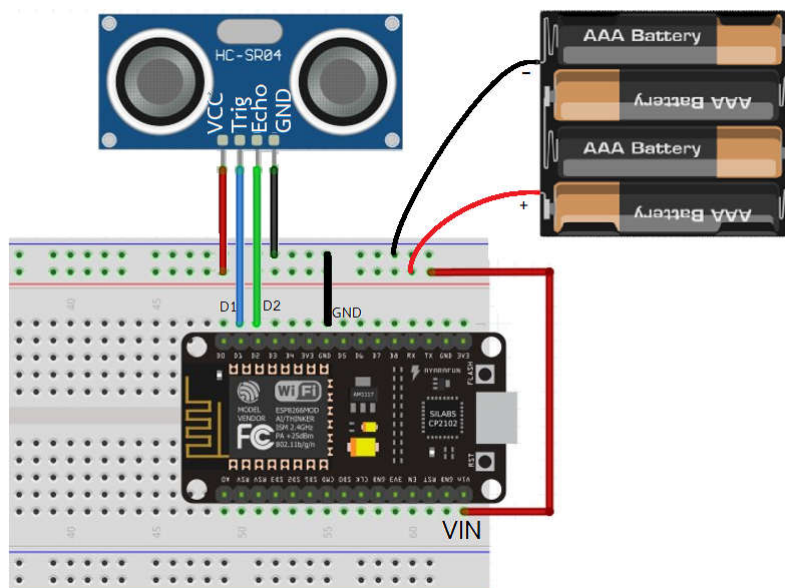
- ให้ผู้เรียนต่อวงจรโมดูลอัลตราโซนิกและติดตั้งไว้บนหุ่นยนต์ให้เรียบร้อย
- เขียนโปรแกรมเพื่ออ่านค่าและแสดงผลบน Blynk ด้วย LCD Display

### เครื่องมือและอุปกรณ์

- |  |               |
|--|---------------|
| ○ บอร์ด NodeMCU/ESP8266 V.2                                    | 1 ตัว         |
| ○ คอมพิวเตอร์ (แบบตั้งโต๊ะหรือพกพา) ที่มีอุปกรณ์รับสัญญาณ Wifi | 1 เครื่อง     |
| ○ โมดูลเซนเซอร์อัลตราโซนิก                                     | 1 ตัว         |
| ○ สมาร์ทโฟน (Android/iOS) พร้อมสัญญาณ 3G                       | 1 เครื่อง     |
| ○ แบตเตอรี่ขนาด AA 4 ก้อนพร้อมที่ใส่                           | 1 ชุด         |
| ○ สายต่อชนิดตัวผู้ทั้งสองด้าน                                  | เท่าที่จำเป็น |

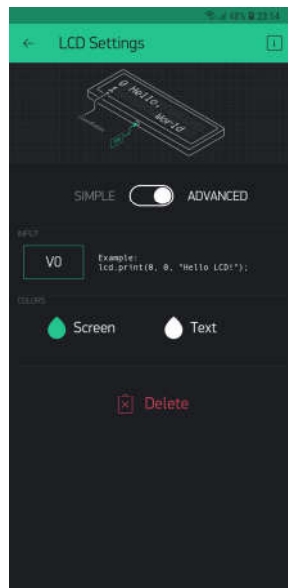
### ลำดับขั้นตอนการทำงาน

1. ให้ต่อวงจรตามรูปที่ 1



รูปที่ 1 วงจรสำหรับกิจกรรมที่ 1

2. สร้าง LCD Display และตั้งค่า Input เป็น V0 ใน Blynk ตามรูปที่ 2



รูปที่ 2 LCD จำลองใน Blynk

3. เขียนโค้ดตามในรูปที่ 3 ลงใน Arduino IDE

```

1 #include <Ultrasonic.h> //เรียกใช้ไลบรารีUltrasonic.h
2 Ultrasonic ultrasonic(D0,D1); //ฟังก์ชันกำหนดขา Tric ที่ขา D0 Echo ที่ขา D1
3 int distanceCM; //ตัวแปรเก็บค่าระยะทางเป็นเซนติเมตร
4 int distanceINC; //ตัวแปรเก็บค่าระยะทางเป็นนิ้ว
5 #define BLYNK_PRINT Serial //ฟังก์ชันการแสดงผลการเชื่อมต่อของ blynk
6 #include <ESP8266WiFi.h> //เรียกใช้ไลบรารีESP8266WiFi
7 #include <BlynkSimpleEsp8266.h> //เรียกใช้ไลบรารีBlynkSimpleEsp8266
8 char auth[] = " "; //หมายเลขที่เคเนที่ได้รับจากการสร้างโปรเจคในblynk
9 char ssid[] = " "; //Username ของ wifi ที่ต้องการเชื่อมต่อ
10 char pass[] = " "; //Password ของ wifi ที่ต้องการเชื่อมต่อ
11 WidgetLCD lcd(V0); //เรียกใช้ LCD ต่อที่ขาจำลอง V0
12 void setup()
13 {
14   Blynk.begin(auth, ssid, pass); //กำหนดค่าการเชื่อมต่อด้วยIPAddress และPort
15   lcd.clear(); //คำสั่งล้างหน้าจอLCD
16   Serial.begin(9600); //กำหนดความถี่ซีเรียลพอร์ต
17 }
18 void loop()
19 {
20   distanceCM = ultrasonic.read(CM); //รับค่าจากอัลตราซันิกมาแปลงเป็นเซนติเมตรเก็บไว้ในตัวแปร
21   distanceINC = ultrasonic.read(INC); //รับค่าจากอัลตราซันิกมาแปลงเป็นนิ้วเก็บไว้ในตัวแปร
22   lcd.print(0,0,"DistanceCM : "); //แสดงข้อความ
23   lcd.print(12,0,distanceCM); //แสดงค่าจากเซ็นเซอร์เป็นเซนติเมตร
24   lcd.print(0,1,"DistanceINC: "); //แสดงข้อความ
25   lcd.print(12,1,distanceINC); //แสดงค่าจากเซ็นเซอร์เป็นนิ้ว
26 }

```

รูปที่ 3 โค้ดของกิจกรรมที่ 1

4. เชื่อมต่อกับแอปพลิเคชัน Blynk

5. ทำการทดลองและสังเกตผลการทดลอง

## 8.2 กิจกรรมที่ 2: หุ่นยนต์หยุดเดินเมื่อเจอสิ่งกีดขวาง

### โจทย์กิจกรรม

ให้ผู้เรียนเขียนโปรแกรมหุ่นยนต์หยุดเดินเมื่อเจอสิ่งกีดขวาง

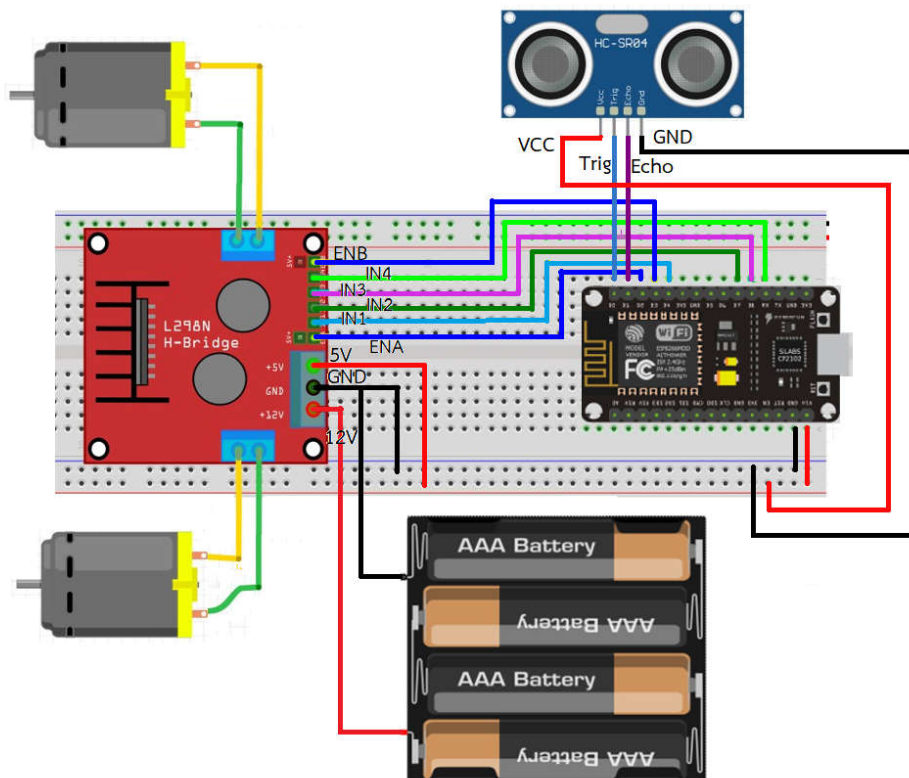
- ให้ผู้เรียนต่อวงจรขับเคลื่อนหุ่นยนต์และอังตราโซนิก
- เขียนโปรแกรมเพื่อควบคุมหุ่นยนต์

### เครื่องมือและอุปกรณ์

- |  |               |
|--|---------------|
| ○ บอร์ด NodeMCU/ESP8266 V.2                                    | 1 ตัว         |
| ○ คอมพิวเตอร์ (แบบตั้งโต๊ะหรือพกพา) ที่มีอุปกรณ์รับสัญญาณ Wifi | 1 เครื่อง     |
| ○ โมดูลเซนเซอร์อัลตราโซนิก                                     | 1 ตัว         |
| ○ สมาร์ทโฟน (Android/iOS) พร้อมสัญญาณ 3G                       | 1 เครื่อง     |
| ○ หุ่นยนต์ที่ประกอบเสร็จเรียบร้อยแล้วจากกิจกรรมก่อนหน้านี้     | 1 ชุด         |
| ○ แบตเตอรี่ขนาด AA 4 ก้อนพร้อมที่ใส่                           | 1 ชุด         |
| ○ สายต่อชนิดตัวผู้ทั้งสองด้าน                                  | เท่าที่จำเป็น |

### ลำดับขั้นตอนการทำงาน

1. ต่อวงจรตามรูปที่ 4



รูปที่ 4 วงจรสำหรับกิจกรรมที่ 2

## 2. เขียนโค้ดตามรูปที่ 5

```

1 #include <Ultrasonic.h> //เรียกใช้ไลบรารี Ultrasonic.h
2 Ultrasonic ultrasonic(D0, D1 ); //ฟังก์ชันกำหนดขา Tric ที่ขา D0 Echo ที่ขา D1
3 int distanceCM; //ตัวแปรเก็บค่าระยะทางเป็นเซนติเมตร
4 int distanceINC; //ตัวแปรเก็บค่าระยะทางเป็นนิ้ว
5 #define PWM_ML D2 //สร้างตัวแปร PWM สำหรับมอเตอร์ซ้าย ต่อกับขา D2
6 #define in1_ML D4 //สร้างตัวแปร in1 สำหรับมอเตอร์ซ้าย ต่อกับขา D4
7 #define in2_ML D7 //สร้างตัวแปร in2 สำหรับมอเตอร์ซ้าย ต่อกับขา D7
8 #define in1_MR D8 //สร้างตัวแปร in1 สำหรับมอเตอร์ขวา ต่อกับขา D8
9 #define in2_MR D9 //สร้างตัวแปร in2 สำหรับมอเตอร์ขวา ต่อกับขา D9
10 #define PWM_MR D3 //สร้างตัวแปร PWM สำหรับมอเตอร์ขวา ต่อกับขา D3
11 int Speed = 767; //สร้างตัวแปร Speed สำหรับกำหนดความเร็วด้วย Duty cycle 75%
12 void setup()
13 {
14   pinMode(PWM_ML, OUTPUT); //กำหนดโหมดการทำงานเอาต์พุต
15   pinMode(in1_ML, OUTPUT);
16   pinMode(in2_ML, OUTPUT);
17   pinMode(PWM_MR, OUTPUT);
18   pinMode(in1_MR, OUTPUT);
19   pinMode(in2_MR, OUTPUT);
20 }
21 void loop()
22 {
23   distanceCM = ultrasonic.read(CM); //รับค่าจากอัลตราซาว์กมาแปลงเป็นเซนติเมตรเก็บไว้ในตัวแปร
24   if (distanceCM <= 10) //เงื่อนไขหากระยะทางน้อยกว่าเท่ากับ 10 เซนติเมตร
25   {
26     Stop(); //ให้หุ่นยนต์หยุด
27   }
28   else //เงื่อนไขหากไม่ใช้
29   {
30     ForWard(); //ให้หุ่นยนต์เดินหน้า
31   }
32 }
33 void ForWard() //ฟังก์ชันเดินหน้า
34 {
35   //สั่งให้มอเตอร์ซ้าย เดินหน้า
36   digitalWrite(in1_ML, HIGH); //สั่งงานให้ in1 สำหรับมอเตอร์ซ้าย เป็น HIGH
37   digitalWrite(in2_ML, LOW); //สั่งงานให้ in2 สำหรับมอเตอร์ซ้าย เป็น LOW
38   analogWrite(PWM_ML, Speed); //สร้างสัญญาณ PWM สำหรับมอเตอร์ซ้าย
39   //สั่งให้มอเตอร์ขวา เดินหน้า
40   digitalWrite(in1_MR, HIGH); //สั่งงานให้ in1 สำหรับมอเตอร์ขวา เป็น HIGH
41   digitalWrite(in2_MR, LOW); //สั่งงานให้ in2 สำหรับมอเตอร์ขวา เป็น LOW
42   analogWrite(PWM_MR, Speed); //สร้างสัญญาณ PWM สำหรับมอเตอร์ขวา
43 }
44 void Stop() //ฟังก์ชันหยุดเคลื่อนที่
45 {
46   //สั่งให้มอเตอร์ซ้าย หยุด
47   digitalWrite(in1_ML, LOW); //สั่งงานให้ in1 สำหรับมอเตอร์ซ้าย เป็น LOW
48   digitalWrite(in2_ML, LOW); //สั่งงานให้ in2 สำหรับมอเตอร์ซ้าย เป็น LOW
49   analogWrite(PWM_ML, 0); //สร้างสัญญาณ PWM เป็น 0 เพื่อ หยุด
50   //สั่งให้มอเตอร์ขวา หยุด
51   digitalWrite(in1_MR, LOW); //สั่งงานให้ in1 สำหรับมอเตอร์ขวา เป็น LOW
52   digitalWrite(in2_MR, LOW); //สั่งงานให้ in2 สำหรับมอเตอร์ขวา เป็น LOW
53   analogWrite(PWM_MR, 0); //สร้างสัญญาณ PWM เป็น 0 เพื่อ หยุด
54 }

```

รูปที่ 5 โค้ดสำหรับกิจกรรมที่ 2

3. ทดลองให้หุ่นยนต์เดินบนพื้นเรียบ แล้วนำวัตถุไปวางด้านหน้าโมดูลเซนเซอร์และสังเกตผลการทำงาน