	สาขาวิชา	อิเล็กทรอนิกส์	**ใบงานการทดลองที่ 14**
	ชื่อวิชา	ไมโครคอนโทรลเลอร์	
	รหัสวิชา	20105-2105	หน้าที่ 108
	ชื่องาน	งานโปรแกรมวัดระยะด้วยโมดูลอัลตร้าโซนิค HC-SR04	

คำชี้แจง ให้ผู้เรียนทุกคนทำการทดลองตามใบงานการทดลองที่ 14 เรื่องโปรแกรมวัดวัดระยะด้วยโมดูลอัลตร้าโซนิค HC-SR04 ตามขั้นตอนการปฏิบัติงาน

จุดประสงค์ทั่วไป

เพื่อให้มีทักษะการปฏิบัติงานโปรแกรมวัดระยะด้วยโมดูลอัลตร้าโซนิค HC-SR04

จุดประสงค์การเรียนรู้เชิงพฤติกรรม (เพื่อให้ผู้เรียน.....)

1. สามารถใช้โปรแกรม Arduino IDE ในการเขียนโปรแกรมภาษา C เบื้องต้นได้อย่างถูกต้อง
2. สามารถใช้งานไมโครคอนโทรลเลอร์ บอร์ด Arduino UNO R3 เบื้องต้นได้อย่างถูกต้อง
3. สามารถประกอบและทดสอบวงจรวัดวัดระยะด้วยโมดูลอัลตร้าโซนิค HC-SR04ได้อย่างถูกต้อง
4. สามารถเขียนโปรแกรมควบคุมการวัดวัดระยะด้วยโมดูลอัลตร้าโซนิค HC-SR04ได้อย่างถูกต้อง
5. สามารถประยุกต์ใช้งานไมโครคอนโทรลเลอร์บอร์ด Arduino UNO R3 เบื้องต้นได้อย่างถูกต้อง
6. มีกิจนิสัยในการแสวงหาความรู้เพิ่มเติม การทำงานด้วยความประณีต รอบคอบและปลอดภัย

เครื่องมือและอุปกรณ์

1.	โปรแกรม Arduino IDE 1.8.4 หรือสูงกว่า	1	โปรแกรม
2.	สาย USB สำหรับ Arduino Uno R3	1	เส้น
3.	ชุดทดลอง Arduino Uno R3 พร้อมสายต่อวงจร	1	ชุด
4.	เครื่องคอมพิวเตอร์แบบพกพา	1	เครื่อง
5.	แผงต่อวงจร	1	ตัว
6.	มัลติมิเตอร์	1	ตัว
7.	เครื่องมือประจำตัว	1	ชุด

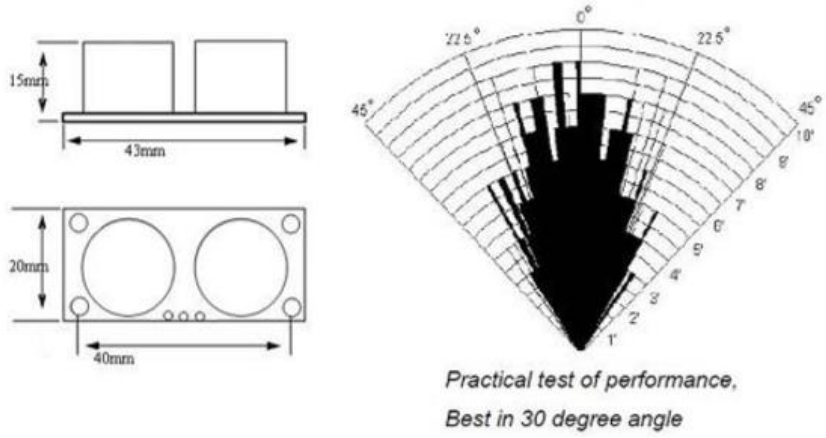
ข้อห้ามและข้อควรระวัง

1. ไม่เล่นและหยอกล้อกันในเวลาปฏิบัติงาน
2. ควรระวังไม่วางบอร์ด Arduino Uno R3 หรือชีลต่างๆ บนโต๊ะโลหะหรือที่วางที่เป็นโลหะ เพราะอาจเกิดการลัดวงจรของภาคจ่ายไฟได้
3. ไม่ควรต่อสายต่อวงจรในบอร์ด Arduino Uno R3 ทิ้งไว้ ควรถอดสายต่อวงจรออกให้หมด เพราะผล การทดลองอาจเกิดการผิดพลาดไม่เป็นไปตามทฤษฎีได้
4. ไม่ควรถอดสายสายโหลด USB เข้าออกตลอดเวลา เพราะอาจทำให้ภาคจ่ายไฟของบอร์ด Arduino Uno R3 เสียหายได้
5. ควรระวังเครื่องมือและอุปกรณ์เสียหายจากการปฏิบัติงานไม่ถูกต้องตามขั้นตอนและไม่ปลอดภัย

PATTAYATECH

ทฤษฎี

เสียงเป็นคลื่นชนิดหนึ่งที่สามารถเดินทางผ่านอากาศได้ดังนั้นถ้าใช้คุณสมบัตินี้มาสร้างเครื่องวัด ระยะทางโดยอาศัยการเดินทางของเสียงผ่านอากาศก็สามารถทำได้เช่นกัน เนื่องจากความถี่เสียงจะมีช่วง ของความถี่ช่วงหนึ่งที่มนุษย์สามารถได้ยินเสียงได้คืออยู่ในช่วง 20Hz-20kHz ดังนั้นหากใช้เสียงในช่วงนี้ ก็จะเป็นการรบกวนการได้ยินของมนุษย์ด้วย ดังนั้นจึงต้องใช้ความถี่ที่สูงกว่าความถี่เสียงปกติเพื่อ หลีกเลี่ยงการได้ยินของมนุษย์ในปัจจุบันได้มีการใช้ความถี่ 40kHz เพื่อใช้ในการวัดระยะทางความถี่นี้ เป็นความถี่ที่สูงกว่าเสียงที่มนุษย์ได้ยินจึงเรียกเสียงนี้ว่า “อัลตร้าโซนิค” โมดูลวัดระยะทางโดยใช้อัลตร้าโซนิคจะประกอบด้วยลำโพงที่ส่งคลื่นเสียงและไมโครโฟนที่ทำหน้าที่เป็นตัวรับเสียง แต่เนื่องจากตัวลำโพงในโมดูลมีขนาดเล็กเสียงที่ส่งออกจึงมีเสียงเบาทำให้ระยะของการวัดไม่ไกลนักซึ่งจะมีระยะไม่ เกิน 5 เมตรและมีมุมที่ใช้งานแคบดังรูป



รูปที่ 14.1 แสดงกราฟเปรียบเทียบคาแรงดันกับระยะทาง

โมดูลอลัตราโซนิคสำหรับวัดระยะทางถูกสร้างมีให้เลือกใช้หลายรุ่นแต่ละรุ่นมีข้อแตกต่างกันอย่างเดียวคือความแม่นยำของการวัดดังตาราง การเลือกใช้งานหากระดับความแม่นยำไม่สำคัญมากนัก โมดูล HR-SC04 จึงเหมาะที่จะนำมาทดลองใช้ในใบงานเนื่องจากเป็นโมดูลวัดระยะที่มีราคาถูกกว่ารุ่นอื่นๆ

ตารางที่ 14.1 แสดงคุณสมบัติของอัลต้าโซนิคแต่ละรุ่น

	HR-SC04	HY-SRF05	US-100
1. Sensor angle	< 15 degrees	< 15 degrees	< 15 degrees
2. Detection distance	2cm-450cm	2cm-450cm	2cm-450cm
3. precision	~3 mm	~2 mm	~1 mm
4. Working Voltage	5VDC	5VDC	5VDC
5. Static current	< 2mA	< 2mA	< 2mA

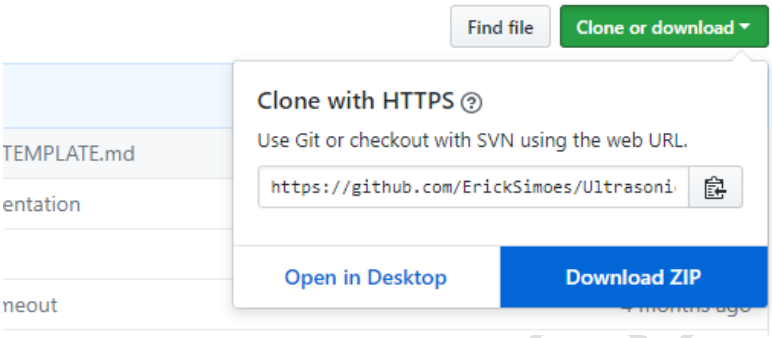
การเขียนโค้ดโปรแกรมเพื่อวัดระยะทางด้วยโมดูลอัลต้าโซนิคสามารถเขียนโดยไม่พึงไลบรารี่ก็ ได้ หรือถ้าหากต้องการใช้ไลบรารี่เพื่อให้การเขียนโค้ดง่ายขึ้นสามารถดาวน์โหลดได้ที่

ตารางที่ 14.2 แสดงแหล่งดาวน์โหลดไลบรารี่สำหรับอัลต้าโซนิค

ไลบรารี่	แหล่งดาวน์โหลด
Ultrasonic.h	https://github.com/ErickSimoes/Ultrasonic

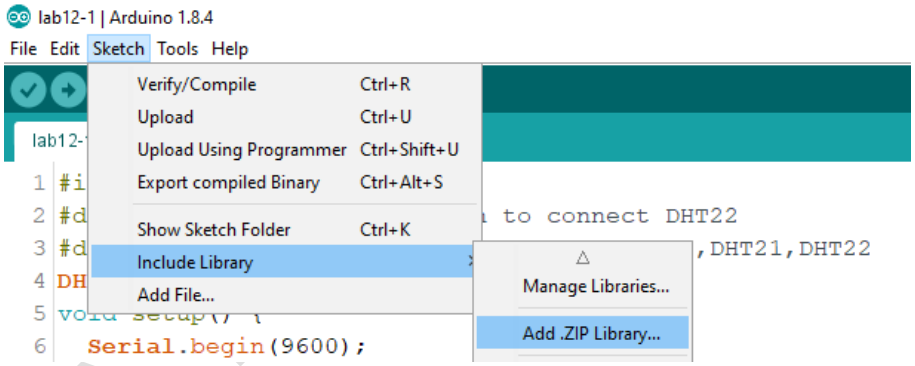
มีขั้นตอนการดำเนินการเพื่อนำไลบรารี่มาใช้งานดังนี้

1. ดาวน์โหลดไลบรารี่ซึ่งเป็นไฟล์ Zip ดังรูป



รูปที่ 14.2 แสดงการดาวน์โหลดไลบรารี่ที่นำมาใช้งาน

2. ทำการเพิ่มไลบรารี่ลงในโปรแกรม Arduino IDE โดยการเพิ่มจากไฟล์ zip แล้วทำการหาไฟล์ zip ที่ได้จากการดาวน์โหลดในข้อ 1



รูปที่ 14.3 แสดงการเพิ่มไลบรารี่ที่เป็นไฟล์ zip ลงในโปรแกรม Arduino IDE

ฟังก์ชั่น Arduino ที่ใช้งานในใบงานการทดลอง

1. **ฟังก์ชั่นกำหนดโหมดการทำงานให้กับขาพอร์ต** โดยสามารถกำหนดได้ทั้งขาดิจิทัลโดยใส่เพียงตวัเลขของขา (0, 1, 2,...13) และขาแอนาลอกที่ต้องการให้ทำงานในโหมดดิจิทัลแต่ การใส่ขาต้องใส่ A นำ หน้าซึ่งใช้ได้เฉพาะ A0, A1,...A5 ส่วนขา A6 และ A7 ไม่สามารถใช้งานในโหมดดิจิทัลได้ รูปแบบของฟังก์ชั่นเป็นดังนี้

pinMode(pin,mode);

pin : หมายเลขขาที่ต้องการเซตโหมด,mode : INPUT, OUTPUT, INPUT_PULLUP

2. ฟังก์ชั่นส่งค่าลอจิกดิจิทัลไปยังขาพอร์ต ค่า HIGH เป็นการส่งลอจิก 1 และค่า LOW เป็นการ ส่งลอจิก 0 ออกไปยังขาพอร์ต ฟังก์ชั่นนี้จะทำงานได้ต้องมีการใช้ฟังก์ชั่น pinMode ก่อน รูปแบบของฟังก์ชั่นเป็นดังนี้

digitalWrite(pin,value);

pin : หมายเลขขาที่ต้องการเขียนลอจิกออกพอร์ต ,value : HIGH หรือ LOW

3. ฟังก์ชั่นอ่านค่าลอจิกดิจิทัลที่ขาพอร์ต เป็นการอ่านค่าเข้ามาซึ่งอาจนำมาเก็บไว้ในตัวแปรไว้ตรวจสอบลอจิกทีหลังหรือจะตรวจสอบลอจิกแบบทันทีก็ได้ ฟังก์ชั่นนี้จะทำงานได้ต้องมี การใช้ฟังก์ชั่น pinMode ก่อน รูปแบบของฟังก์ชั่นเป็นดังนี้

digitalRead(PIN); pin : หมายเลขขาพอร์ตที่ต้องการอ่านลอจิก

ตัวอย่างเช่น value=digitalRead(2); หมายถึง อ่านค่าลอจิกที่ขา D2 มาเก็บไว้ในตัวแปร value if(digitalRead(2)==LOW) หมายถึง ตรวจสอบขา D2 ว่าเป็นลอจิก 0 หรือไม่

4. ฟังก์ชั่นหน่วงเวลาหรือฟังก์ชั่นหยุดค้าง การใช้งานสามารถกำหนดตัวเลขของเวลาที่ต้องการหยุดค้าง ตัวเลขที่ใส่เป็นตัวเลขของเวลาหน่วยเป็นมิลลิวินาที ตัวเลขของเวลาที่ใส่ ได้สูงสุดคือ 4,294,967,295 ซึ่งเป็นขนาดของตวัแปร unsigned long รูปแบบของฟังก์ชั่นเป็นดังนี้

Delay(ms); ms : ตัวเลขที่หยุดค้างของเวลาหน่วยมิลลิวินาที (unsigned long)

5. ฟังก์ชั่นกำหนดความเร็วในการสื่อสารทางพอร์ตอนุกรม รูปแบบของฟังก์ชั่นเป็นดังนี้

Serial.begin(speed); speed: ตัวเลขของอัตราเร็วในการสื่อสารผ่านพอร์ตอนุกรม

6. ฟังก์ชั่นส่งข้อมูลออกพอร์ต เป็นฟังก์ชั่นที่ใช้ในการส่งข้อมูลออกทางพอร์ตอนุกรมหรือพิมพ์ข้อมูลออกทางพอร์ตเพื่อแสดงผลที่จอคอมพิวเตอร์ เมื่อพิมพ์เสร็จตัวเคอร์เซอร์จะรออยู่ที่ท้ายสิ่งที่พิมพ์นั้น ๆ รูปแบบของฟังก์ชั่นเป็นดังนี้

Serial.print(val); Serial.print(val, format);

7. ฟังก์ชั่นส่งข้อมูลออกพอร์ต คล้ายกับฟังก์ชั่น Serial.print ต่างกันตรงที่เมื่อพิมพ์เสร็จตัวเคอร์เซอร์จะขึ้นมารอยังบรรทัดใหม่ ดังนั้นเมื่อสั่งพิมพ์ครั้งถัดไปข้อมูลที่ปรากฏจะอยู่ที่บรรทัดใหม่แทนที่จะต่อท้ายเหมือนกับฟังก์ชั่น Serial.print รูปแบบของฟังก์ชั่นเป็นดังนี้

Serial.println(val); Serial.println(val, format);

8. ฟังก์ชั่นวัดความกว้างของพัลซ์ โดยค่าที่วัดได้เป็นเวลาหน่วยเป็นไมโครวินาที (uS) ผู้ใช้งานสามารถระบุลอจิกของสัญญาณที่ใช้ในการวัดได้ เช่นเมื่อกำหนดลอจิกที่ใช้ตรวจจับเป็น HIGH ฟังก์ชั่นจะเริ่มนับเวลาเมื่อขาสัญญาณที่ตรวจจับเป็นลอจิก HIGH โดยนับเวลาไป จนกว่าสัญญาณจะเป็นเป็น LOW เวลาที่ได้เป็นตัวเลขที่มีหน่วยเป็นไมโครวินาที รูปฟังชั่น เป็นดังนี้

pulseIn(pin, value);

pin: ขาพอร์ตที่ใช้ในการตรวจจบัสัญญาณพลัซ์

value: ค่าลอจิกที่ใช้ก าหนดเพื่อใช้ในนับเวลาโดยฟังก์ชั่นจะหยุดนับเมื่อค่า ลอจิกเป็นตรงข้าม

ตัวอย่างเช่น duration = pulseIn(12, HIGH);

หมายถึง วัดความกว้างของพัลซ์โดยการจับเวลา เมื่อขา D12 เป็นลอจิก HIGH จนกระทั่งขา D12 มีการเปลี่ยนลอจิกเป็น LOW โดยค่าที่ได้เป็นตัวเลขหน่วยเป็น ไมโครวินาทีแล้วเอาไปเก็บไว้ในตัวแปร duration

ฟังก์ชั่นใช้งานของไลบรารี่ Ultrasonic.h

การอ่านค่าจากโมดูล Ultrasonic จำเป็นต้องใช้ไลบรารี่ช่วยงาน ซึ่งไลบรารี่ไม่ได้ถูกเพิ่มเข้ามาใน ตัวโปรแกรม Arduino IDE ตั้งแต่แรกจำเป็นต้องติดตั้งเพิ่มเติม โดยมีฟังก์ชั่นให้ใช้งานดังนี้

1. **ฟังก์ชั่นกำหนดขาเชื่อมต่อ** ใช้ในการระบุขาที่ใช้เชื่อมต่อให้ตัวโปรแกรมรับรู้ รูปแบบดังนี้

   Ultrasonic ultrasonic(Trig PIN,Echo PIN);

   Trig PIN : ตัวเลขระบุขาพอร์ตที่ใช้เชื่อมต่อกับขา Trig ของโมดูล

   Echo PIN : ตัวเลขระบุขาพอร์ตที่ใช้เชื่อมต่อกับขา Echo ของโมดูล

   ตัวอย่างเช่น Ultrasonic ultrasonic(9,8); หมายถึง ต่อไปในโปรแกรมจะใช้ชื่อ ultrasonic ในการเรียกใช้งานโมดูล โดยมีการ เชื่อมต่อขาพอร์ต D9 เข้าที่ขา Trig และขาพอร์ต D8 เข้าที่ขา Echo ของโมดูล

2. **ฟังก์ชั่นอ่านค่าระยะทาง** ใช้อ่านค่าระยะทางจากตัวโมดูลโดยสามารถระบุหน่วยที่ต้องการวัดได้ 2 แบบคือ เซนติเมตร (CM) และหน่วยที่เป็นนิ้ว (INC) รูปแบบเป็นดังนี้

   .distanceRead();

   Unit : หน่วยที่ต้องการวัด CM, INC

   ตัวอย่างเช่น Serial.print(ultrasonic. distanceRead (CM));

   หมายถึง แสดงผลระยะที่วัดได้หน่วยเป็นเซนติเมตรทางพอร์ตอนุกรม

การเปลยี่นค่า time out เพื่อปรับระยะการวัด (สำหรับเมื่อใช้งานไลบรารี่)

ค่าเริ่มต้นของไลบรารี่ก าหนดค่า time out ไว้ที่ 20000UL ทำให้สามารถวัดระยะได้ไกลสุดประมาณ 3.4 เมตร การใช้ฟังก์ชั่นที่มีการป้อนค่า 2 ค่าที่เป็นขาเชื่อมต่อดังนี้

Ultrasonic ultrasonic(Trig PIN,Echo PIN);

Trig PIN : ขา Trig ของโมดูลอลัตร้าโซนิค

Echo PIN : ขา Echo ของโมดูลอลัตร้าโซนิค

ผู้ใช้สามารถเปลี่ยนค่าระยะการวัดได้ โดยการกำหนดค่า TimeOut เข้าในฟังก์ชั่น ตัวอย่างเช่น

Ultrasonic ultrasonic(Trig PIN,Echo PIN,MaxTimeout);

Trig PIN : ขา Trig ของโมดูลอลัตร้าโซนิค

Echo PIN : ขา Echo ของโมดูลอลัตร้าโซนิค

Max.TimeOut : ค่าเวลา Time out สูงสุดสำหรับวัดระยะที่ต้องการวัด (µs)

ตัวอย่างเช่น Ultrasonic ultrasonic(9,8,29000UL);

หมายถึง ใช้ชื่อ ultrasonic แทนตัวเซนเซอร์ ขา Trig ของเซนเซอร์เชื่อมต่อเข้าที่ขา D9 ขา Echo ของเซนเซอร์เชื่อมต่อเข้าที่ขา D8 ค่า TimeOut เท่ากับ 29000 µs ซึ่ง สามารถวัดระยะได้สูงสุดประมาณ 5 เมตร

PATTAYATECH

ตารางที่ 14.3 แสดงการวิธีการคำนวณค่า TimeOut เพื่อให้ได้ระยะการวัดที่ต้องการ

ระยะการสูงสุดที่ต้องการวัด	สูตร	ตัวอย่างการคำนวณ
หน่วยเป็นเซนติเมตร	TimeOut=Max.Distance(cm) * 58	500 cm * 58 = 29000 µs
หน่วยเป็นนิ้ว	TimeOut=Max.Distance(inc) * 148	25 inc * 148 = 3700 µs

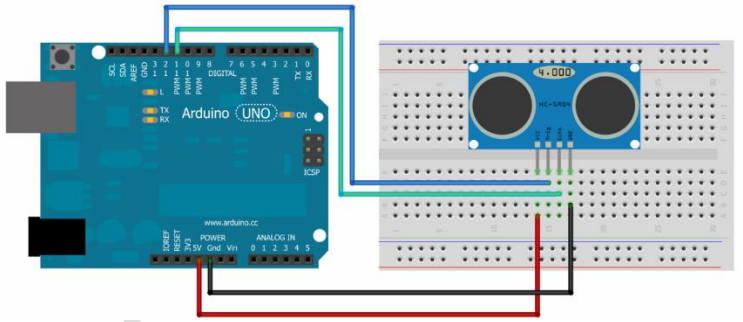
หมายเหตุ ค่า Time out ไม่ควรเกินค่าระยะทางที่โมดูลอัลตร้าโซนิคทำงานได้นั่นคือ 450 cm [ที่มา:ครูประภาส สุวรรณเพชร,เอกสารประกอบการอบรม เรียนรู้และลองเล่น Arduino เบื้องต้น **(ฉบับปรับปรุงครั้งที่ 1)** ,หน้าที่ 160-164.]

ลำดับขั้นการทดลอง

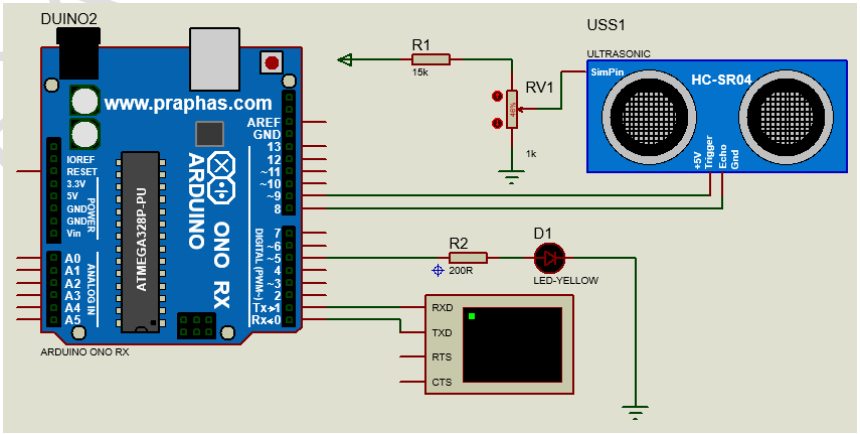
ตอนที่ 1 เขียนโปรแกรมวัดระยะด้วยโมดูลอัลตร้าโซนิค HC-SR04 แบบไม่ใช้ไลบรารี่

แนวคิดการเรียนรู้ คือ เขียนโปรแกรมวัดระยะโดยการใช้โมดูลอัลตร้าโซนิคแบบไม่ใช้ไลบรารี่ แสดงผลที่จอคอมพิวเตอร์ผ่านทางพอร์ตอนุกรมโดยมีขั้นตอนดังนี้

1. ประกอบวงจรการวัดระยะด้วยโมดูลอัลตร้าโซนิค HC-SR04 ใช้บอรด Arduino UNO R3 ดังรูปที่ 14.4



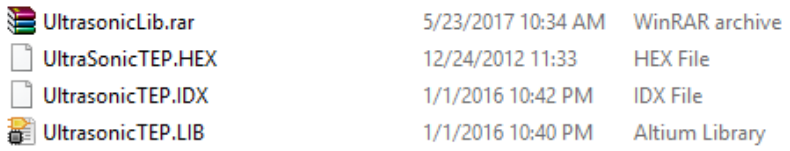
(ก) วงจรไมโครคอนโทรลเลอร์ที่ใช้บอร์ด Arduino Uno



(ข) การต่อวงจรทดลองในโปรแกรมจำลองการทำงาน

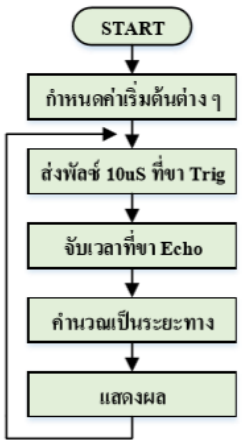
รูปที่ 14.4 แสดงการต่อวงจรการวัดระยะด้วยโมดูลอัลตร้าโซนิค HC-SR04

เนื่องจากโปรแกรมจำลองการทำงานไม่มีโมเดลอัลตราโซนิครุ่น HR-SC04 ให้จำลองจึงต้องใช้ โมเดลจากไลบรารี่ที่เพิ่มเข้ามา ซึ่งไลบรารี่ดังกล่าวภายในเป็นซีพียูที่เขียนโค้ดให้ส่งค่าเลียนแบบอัลตร้า โซนิคซึ่งสามารถปรับค่าได้โดยอาศัยแรงดันควบคุมผ่านการปรับโพเทนธิโอมิเตอร์ ดังนั้นเมื่อใช้งาน จะต้องโหลดไฟล์โปรแกรมควบคุมเข้าโมเดลเช่นเดียวกับการจำลองไมโครคอนโทรลเลอร์ ไฟล์ที่ต้องโหลดเข้าโมเดลเป็นไฟล์ภาษาเครื่องที่มาพร้อมกับไลบรารี่ดังรูป



รูปที่ 14.5 แสดงไฟล์ภาษาเครื่องสำหรับโมเดลอัลตร้าโซนิค

2. เปิดโปรแกรม Arduino IDE จากนั้นพิมพ์โค้ดโปรแกรมวัดระยะด้วยโมดูลอัลตร้าโซนิค HC-SR04โดยใช้บอรด Arduino UNO R3 ตามรูปที่ 14.6 ดังต่อไปนี้



(ก) ผังงาน

```
#define TRIGGER_PIN  9 //  digital pin D9
#define ECHO_PIN     8 //  digital pin D8
void setup()
{
  Serial.begin (9600);
  pinMode(TRIGGER_PIN, OUTPUT);
  pinMode(ECHO_PIN, INPUT);
}
void loop()
{
  double duration, distance;
  digitalWrite(TRIGGER_PIN, LOW);          // Get
  delayMicroseconds(2);                    // stab
  digitalWrite(TRIGGER_PIN, HIGH);         // send
  delayMicroseconds(10);                   // dela
  digitalWrite(TRIGGER_PIN, LOW);          // afte
  duration = pulseIn(ECHO_PIN, HIGH);      // calcu
  distance = (duration/2) / 29.1;          // sing
  Serial.print(distance);Serial.println(" cm");
  delay(500);
}
```

(ข) โค้ดโปรแกรม

รูปที่ 14.6 แสดงโปรแกรมวัดระยะด้วยโมดูลอัลตร้าโซนิค HC-SR04

3. บันทึกไฟล์โค้ด ชื่อ Lab14-1
4. ทำการ Compile โค้ด Lab14-1
5. เชื่อมต่อสาย USB กับ บอร์ด Arduino Uno R3
6. Upload โปรแกรม Lab14-1 ลงบอรด Arduino UNO R3

7. สังเกตวงจรการทำงานและบันทึกผลการทดลอง

.....

.....

.....

.....

.....

.....

.....

8. คำถามท้ายการทดลองตอนที่ 1 จากโค้ดโปรแกรม Lab14-1 จงตอบคำถามต่อไปนี้

8.1. บรรทัดที่ 1,2 ทำหน้าที่.....

8.2. บรรทัดที่ 5 ทำหน้าที่.....

8.3. บรรทัดที่ 6,7 ทำหน้าที่.....

8.4. บรรทัดที่ 11 ทำหน้าที่.....

8.5. บรรทัดที่ 12-16 ทำหน้าที่.....

8.6. บรรทัดที่ 17 ทำหน้าที่.....

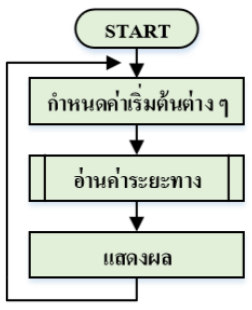
8.7. บรรทัดที่ 18 ทำหน้าที่.....

8.8. บรรทัดที่ 19 ทำหน้าที่.....

ตอนที่ 2 เขียนโปรแกรมวัดระยะด้วยโมดูลอัลตร้าโซนิค HC-SR04 แบบใช้ไลบรารี่

แนวคิดการเรียนรู้ คือ เขียนโปรแกรมวัดระยะโดยการใช้โมดูลอัลตร้าโซนิคแบบใช้ไลบรารี่แสดงผลที่จอคอมพิวเตอร์ผ่านทางพอร์ตอนุกรม โดยมีขั้นตอนดังนี้

9. เปิดโปรแกรม Arduino IDE จากนั้นพิมพ์โค้ดโปรแกรมวัดระยะด้วยโมดูลอัลตร้าโซนิค HC-SR04โดยใช้บอรด Arduino UNO R3 ตามรูปที่ 14.7 ดังต่อไปนี้



(ก) ผังงาน

```
1  #include <Ultrasonic.h>
2  Ultrasonic ultrasonic(9,8); // (Trig PIN,Echo PIN)
3  void setup() {
4    Serial.begin(9600);
5  }
6  void loop()
7  {
8    Serial.print(ultrasonic.distanceRead(CM)); // CM or INC
9    Serial.println(" cm" );
10   delay(100);
11 }
```

(ข) โค้ดโปรแกรม

รูปที่ 14.7 แสดงโปรแกรมวัดระยะด้วยโมดูลอัลตร้าโซนิค HC-SR04

10. บันทึกไฟล์โค้ด ชื่อ Lab14-2
11. ทำการ Compile โค้ด Lab14-2
12. เชื่อมต่อสาย USB กับ บอร์ด Arduino Uno R3
13. Upload โปรแกรม Lab14-2 ลงบอรด Arduino UNO R3

14. สังเกตวงจรการทำงานและบันทึกผลการทดลอง

.....

.....

.....

.....

15. คำถามท้ายการทดลองตอนที่ 2 จากโค้ดโปรแกรม Lab14-2 จงตอบคำถามต่อไปนี้

15.1. บรรทัดที่ 1 ทำหน้าที่.....

15.2. บรรทัดที่ 2 ทำหน้าที่.....

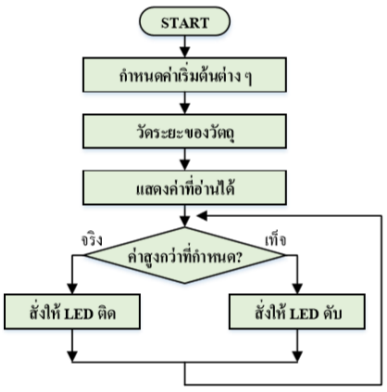
15.3. บรรทัดที่ 4 ทำหน้าที่.....

15.4. บรรทัดที่ 8 ทำหน้าที่.....

ตอนที่ 3 เขียนโปรแกรมควบคุมการติดดับของ LED จากระยะห่างของวัตถุ

แนวคิดการเรียนรู้ คือ เขียนโปรแกรมควบคุมการติดดับของ LED จากระยะห่างของวัตถุที่วัดได้ หากวัตถุวางอยู่ ในระยะต่ำกว่า 30 cm ให้ LED ติดหากระยะของวัตถุเกิน 30 cm ให้ LED ดับ พร้อมแสดงผลระยะที่จอคอมพิวเตอร์ผ่านทางพอร์ตอนุกรม โดยมีขั้นตอนดังนี้

16. เปิดโปรแกรม Arduino IDE จากนั้นพิมพ์โค้ดโปรแกรมวัดระยะด้วยโมดูลอัลตร้าโซนิค HC-SR04โดยใช้บอรด Arduino UNO R3 ตามรูปที่ 14.8 ดังต่อไปนี้




```
1  #include <Ultrasonic.h>
2  #define LED 5
3  Ultrasonic ultrasonic(9,8); // (Trig PIN,Echo PIN)
4  void setup() {
5    Serial.begin(9600);
6    pinMode(LED,OUTPUT);
7  }
8  void loop()
9  {
10   int distance=ultrasonic.distanceRead(CM);
11   Serial.print(distance);
12   Serial.println(" cm" );
13   digitalWrite(LED, distance>30? HIGH:LOW);
14   delay(100);
15 }
```

(ก) ผังงาน (ข) โค้ดโปรแกรม

รูปที่ 14.8 แสดงโปรแกรมควบคุมการติดดับของ LED จากระยะห่างของวัตถุ

17. บันทึกไฟล์โค้ด ชื่อ Lab14-3
18. ทำการ Compile โค้ด Lab14-3
19. เชื่อมต่อสาย USB กับ บอร์ด Arduino Uno R3
20. Upload โปรแกรม Lab14-3 ลงบอรด Arduino UNO R3

21. สังเกตวงจรการทำงานและบันทึกผลการทดลอง

.....

.....

.....

.....

22. คำถามท้ายการทดลองตอนที่ 2 จากโค้ดโปรแกรม Lab14-2 จงตอบคำถามต่อไปนี้

22.1. บรรทัดที่ 1 ทำหน้าที่.....

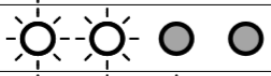
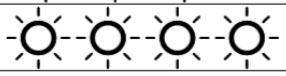
22.2. บรรทัดที่ 3 ทำหน้าที่.....

22.3. บรรทัดที่ 10 ทำหน้าที่.....

22.4. บรรทัดที่ 13 ทำหน้าที่.....

ตอนที่ 4 งานที่มอบหมาย

เขียนโปรแกรมแจ้งเตือนการเข้าใกล้ (เช่นเดียวกับเซนเซอร์ถอยหลังของรถยนต์) แสดงผลที่ LED จำนวน 4 ตัวโดยมีเงื่อนไขการแสดงผลดังนี้

ระยะตรวจจับ	การติดดับของ LED
$s > 1.5m$	
$1.0m \le s \le 1.5m$	
$0.5m \le s \le 1.0m$	
$0s \le 0.5m$	

23. จงเขียนผังงานจากงานที่มอบหมาย

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

24. พิมพ์โค้ดโปรแกรมตามผังงานในข้อที่ 23
25. บันทึกไฟล์โค้ด ชื่อ Lab14-4
26. ทำการ Compile โค้ด Lab14-4
27. เชื่อมต่อสาย USB กับ บอร์ด Arduino Uno R3
28. Upload โปรแกรม Lab14-4 ลงบอรด Arduino UNO R3
29. สังเกตวงจรการทำงานและบันทึกผลการทดลอง

.....

.....

.....

.....

.....

30. สรุปผลการทดลอง

.....

.....

.....

.....

.....

.....

.....

PATTAYATECH