ไม้นำทางแจ้งเตือนสิ่งกีดขวาง

(guide stick for the visually impaired)

จัดทำโดย

นรจ.วันชัย นาคนาคา

นรจ.กฤตพจน์ สายสิม

นรจ.พัฒนะ ทรัพย์สุนทร

นรจ.เฉลิมวงศ์ คำแพง

นรจ.ณัฐพงศ์ ศาสนาไทย

นรจ.ณัฐกฤษฎิ์ จันทร์ประทัด

โครงงานนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรนักเรียนจ่าทหารเรือชั้นปีที่ ๒

พรรคพิเศษ เหล่า ช่างยุทธโยธาอิเล็กทรอนิกส์ ปีการศึกษา ๒๕๖๓

โรงเรียนอิเล็กทรอนิกส์ กองวิทยาการ กรมอิเล็กทรอนิกส์ทหารเรือ

หัวข้อ : ไม้นำทางแจ้งเตือนสิ่งกีดขวาง
โดย : นรจ.วันชัย นาคนาคา
นรจ.กฤตพจน์ สายสิม
นรจ.พัฒนะ ทรัพย์สุนทร
นรจ.เฉลิมวงศ์ คำแพง
นรจ.ณัฐพงศ์ ศาสนาไทย
นรจ.ณัฐกฤษฎิ์ จันทร์ประทัด
ครูที่ปรึกษา : ร.ต.กฤษณะ เพิ่มคำ
พ.จ.อ.อรุณ เจริญศรี
จ.อ.ธนากร ภัทรกุลปรีดา
พรรค : พิเศษ
เหล่า : ช่างยุทธโยธาอิเล็กทรอนิกส์
ปีการศึกษา : ๒๕๖๓
สถานศึกษา : โรงเรียนอิเล็กทรอนิกส์ กองวิทยาการ กรมอิเล็กทรอนิกส์ทหารเรือ

บทคัดย่อ
(Abstract)

มีวัตถุประสงค์เพื่อส่งเสริมการใช้เวลาว่างให้เกิดประโยชน์และสร้างเสริมค่านิยมจิตสาธารณะแก่สังคม ชุมชนและเพื่อพัฒนาอุปกรณ์สำหรับอำนวยความสะดวกแก่ผู้พิการทางสายตาให้สามารถดำรงชีวิตอยู่ในชีวิตประจำวันได้อย่างเป็นปกติสุขโดยใช้บอร์ด Arduino

ผลการทดลองพบว่า ไม้นำทางแจ้งเตือนสิ่งกีดขวางสามารถทำงานได้ตามความพึงพอใจ ของคณะผู้จัดทำโดยเมื่อนำไม้นำทางแจ้งเตือนสิ่งกีดขวางไปทดลองใช้งานจริง ไม้นำทางของทางคณะผู้จัดทำสามารถทำงานได้จริง และสามารถแจ้งเตือนสิ่งกีดขวางได้อย่างมีประสิทธิภาพ

กิตติกรรมประกาศ

(Acknowledgement)

การทำโครงงานเรื่องไม้นำทางแจ้งเตือนสิ่งกีดขวาง ประสบความสำเร็จ ได้ด้วยดีเนื่องจากได้รับความกรุณาจากอาจารย์ที่ปรึกษาโครงงาน ร.ต.กฤษณะ เพิ่มคำ พ.จ.อ.อรุณ เจริญศรี และ จ.อ.ธนากร ภัทรกุลปรีดา ที่ได้ให้ความช่วยเหลือเกี่ยวกับแนวคิด ในการจัดทำโครงงานดูแลเอาใจใส่ติดตามชิ้นงาน อีกทั้งชี้แนะข้อบกพร่องตลอดจนสนับสนุนคณะผู้จัดทำให้มีความสามารถในการทำโครงงานจนเสร็จลุล่วงได้ด้วยดี

ความดีและประโยชน์ขอมอบให้กับครูบาอาจารย์ทุกท่าน ที่ให้การสนับสนุน จนทำให้นักเรียนจ่า มีความเข้าใจและความรู้ จึงส่งผลให้การทำโครงงานชิ้นนี้สำเร็จลุล่วงไปด้วยดี ท้ายนี้คณะผู้จัดทำโครงงานใคร่ ขอกราบขอบพระคุณ บิดา มารดา ซึ่งสนับสนุนในด้านการเงินและให้กำลังใจแก่ผู้จัดทำโครงงาน

นรจ.วันชัย นาคนาคา
นรจ.กฤตพจน์ สายสิม
นรจ.พัฒนะ ทรัพย์สุนทร
นรจ.เฉลิมวงศ์ คำแพง
นรจ.ณัฐพงศ์ ศาสนาไทย
นรจ.ณัฐกฤษฎิ์ จันทร์ประทัด
คณะผู้จัดทำ

สารบัญ

หน้า

บทคัดย่อ ก
กิตติกรรมประกาศ ข
บทที่ 1 บทนำ 1
- ที่มาและความสำคัญ 1
- วัตถุประสงค์ 1
- ขอบเขตการศึกษา 1
- สมมติฐาน 1
- ประโยชน์ที่คาดว่าจะได้รับ 1

บทที่ 2 ทฤษฎีที่เกี่ยวข้อง 2
- Ultrasonic sensor HC-SR04 2
- Arduino Nano v3 6
- Buzzer 10
- แบตเตอรี่ลิเธียมไอออน 11
- มอเตอร์ (Motor) 14
- โปรแกรม Arduino IDE โครงสร้างของการเขียนโปรแกรม Arduino 16
- โปรแกรมควบคุมการทำงาน 19

บทที่ 3 วิธีการดำเนินงาน 24
- วิธีการดำเนินงาน 24
- การทำงานของระบบ 25
- วัสดุอุปกรณ์ 26
- ขั้นตอนการดำเนินงาน 28

บทที่ 4 ผลการดำเนินงาน 32
บทที่ 5 สรุปผลการดำเนินงานและข้อเสนอแนะ 33
บรรณานุกรม 34
ภาคผนวก 35

บทที่ 1
บทนำ

1.1 ที่มาและความสำคัญของโครงงาน

ในปัจจุบันผู้พิการทางสายตามีจำนวนมากซึ่งด้วยสภาพสายตาที่ไม่เอื้ออำนวยจึงส่งผลให้การใช้ชีวิตผิดจากคนปกติจึงมีความต้องการทางด้านเครื่องมือเพื่ออำนวยความสะดวกในการดำเนินชีวิตให้แก่ผู้พิการทางสายตา คือเทคโนโลยีสำหรับช่วยในการเดินทาง ดังนั้นทางคณะผู้จัดทำโครงงานได้ตระหนักถึงปัญหานี้จึงได้ทำให้เกิดโครงงานนี้ขึ้นมาเพื่อให้เป็นสิ่งอำนวยความสะดวกในการเดินทางสำหรับผู้พิการทางสายตา

1.2 วัตถุประสงค์ของโครงงาน

1.เพื่อศึกษาการทำงานของ Ultrasonic sensor และArduino Nano V3

2.เพื่อศึกษาการเขียนโปรแกรมด้วยภาษา C เพื่อใช้ควบคุม Arduino

3.เพื่อออกแบบและสร้างไม้นำทางสำหรับช่วยอำนวยความสะดวกในการเดินทางให้กับผู้พิการทางสายตา

1.3 ขอบเขตการศึกษา

สำหรับโครงงานนี้มีขอบเขตโดยแบ่งการทำงานออกเป็น 2 ส่วนคือ

1. ตรวจจับสิ่งกีดขวาง
   - สามารถตรวจจับสิ่งกีดขวางที่อยู่ข้างหน้าก่อนก่อนผู้ใช้งานจะเดินไปถึง 50 cm
2. ส่วนแสดงผลการแจ้งเตือน
   - ก่อนถึงสิ่งกีดขวาง 50 cm Ultrasonic sensor จะทำงานทำให้บัซเซอร์ทำงานมีเสียงเตือนดังขึ้น
   - เมื่อบัซเซอร์ทำงานจะทำให้มอเตอร์ทำงานทำให้เกิดการสั่นขึ้นทำให้ผู้ใช้งานรู้สึกได้ว่ามีสิ่งกีดขวางอยู่ด้านหน้า

1.4 สมมตติฐาน

เมื่อ Ultrasonic sensor ตรวจพบสิ่งกีดขวางระยะ 50 เซนติเมตร จะทำให้ sensorทำงาน และสั่งให้มอเตอร์สั่น ทำงานทำการสั่นเพื่อให้ผู้พิการทางสายตา ทราบว่ามีสิ่งกีดขวางอยู่ด้านหน้าเพื่อทำการหลบสิ่งกีดขวาง

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1. สามารถอำนวยความสะดวกแก่ผู้พิการทางสายตาได้โดยการเตือนว่าจะชนสิ่งกีดขวาง
2. สามารถนำความรู้ที่ได้มาใช้ในการประกอบวิชาชีพ
3. สามารถนำความรู้ทางด้านทฤษฎีมาประยุกต์ใช้ในทางปฏิบัติ
4. สามารถเป็นแนวทางในการนำเอาไปพัฒนาต่อยอดทางด้านอุตสาหกรรมด้านพาณิชย์

บทที่ 2
ทฤษฎีและหลักการที่เกี่ยวข้อง
ข้อมูลอุปกรณ์ที่ใช้ในอุปกรณ์

2.1 Ultrasonic sensor HC-SR04

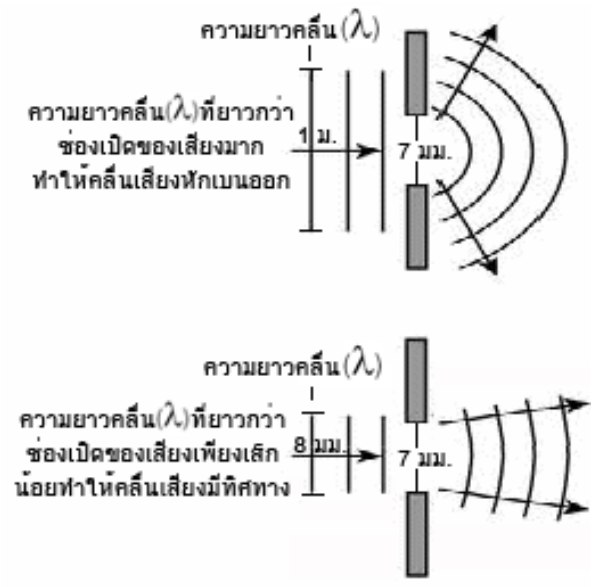
เป็นอุปกรณ์ใช้วัดระยะทางโดยไม่ต้องมีการสัมผัสกับตำแหน่งที่ต้องการวัด วัดได้ตั้งแต่ 2 cm ถึง 400 cm โดยส่งสัญญาณ Ultrasonic ความถี่ 40 kHz ไปที่วัตถุที่ต้องการวัดและรับสัญญาณที่สะท้อนกลับมา พร้อมทั้งจับเวลาเพื่อนำมาใช้ในการคำนวณระยะทาง ข้อดีของ Ultrasonic sensor เมื่อเทียบกับ Photoelectric Sensors แบบใช้วัดระยะทางก็คือ แม้แต่ในสภาวะที่ไม่เอื้ออำนวยเช่นฝุ่นผงและความสกปรก สามารถตรวจวัดระยะห่างของวัตถุได้ดีแม้ว่าวัตถุนั้นจะมีความโปร่งใสโปร่งแสงมีความแวววาวได้อย่างแม่นยำ และยังเหมาะสำหรับการตรวจจับของเหลวและวัตถุที่เป็นเม็ดได้เป็นอย่างดี

อัลตร้าโซนิค (Ultrasonic)

Ultrasonic คือ คลื่นเสียงที่มีความถี่สูงเกิน 20,000 Hz มาก จนหูมนุษย์ไม่สามารถได้ยินได้ คลื่นเสียงที่มนุษย์สามารถได้ยินได้ดีนั้นจะอยู่ที่ประมาณ 20 Hz จนถึง 15 kHz โดยเฉลี่ยสำหรับบุคคลคนที่มีอายุเลยวัยเด็กจะประมาณ 20 Hz จนถึง 20 kHz สำหรับเด็กที่มีอายุน้อยๆแต่คลื่น Ultrasonic จะมีความถี่อยู่ที่ 20 kHz ขึ้นไป ซึ่งมีความถี่ที่สูงจนมนุษย์ไม่สามารถได้ยินได้เลย ซึ่งความถี่นี้จะมีสัตว์อยู่บางประเภทที่สามารถได้ยินได้ เช่น ค้างคาว และ โลมา เนื่องจากค้างคาวมีดวงตาที่เล็กและออกหากินในเวลากลางคืนทำให้ค้างคาวต้องมีสิ่งที่มาทดแทนคือ คลื่นUltrasonicนั่นเองซึ่งค้างคาวจะใช้คลื่นความถี่Ultrasonicเพื่อใช้ระบุตำแหน่ง รูปร่าง ทิศทาง ของวัตถุที่ขวางเส้นการเดินทางและความเร็วในการเคลื่อนที่ของสิ่งๆนั้นได้อย่างแม่นยำ ด้วยหลักการสะท้อนของคลื่นเสียงที่ว่า มุมตกกระทบเท่ากับมุมสะท้อน ด้วยหลักการเดียวกันนี้ มนุษย์จึงนำคลื่นย่าน Ultrasonic นี้มาใช้งาน เนื่องจากเป็นคลื่นที่มีทิศทาง ทำให้สามารถเล็งคลื่นไปที่ตำแหน่งหรือเป้าหมายที่ต้องการได้อย่างแม่นยำ เนื่องจากคุณสมบัติของคลื่นที่ว่า ยิ่งคลื่นมีความถี่สูงมากเท่าไหร่ ความยาวคลื่นก็จะยิ่งสั้นลง

ถ้าความยาวของคลื่นยาวกว่าช่องเปิด หรือช่องที่คลื่นเสียงสามารถออกมาได้เช่น คลื่น 300 Hz จะมีความยาวคลื่นประมาณ 1 เมตร ซึ่งยาวกว่าช่องเปิดที่คลื่นเสียงสามารถออกมาได้ ทำให้คลื่นนั้นเกิดการหักเบนออก ที่ขอบด้านนอกของช่องเปิด ทำให้คลื่นมีการกระจายตัว ไม่สามารถระบุตำแหน่งหรือโฟกัสไปที่ จุดๆหนึ่งได้ แต่ถ้าคลื่น

มีความถี่สูง ความยาวคลื่นก็จะสั้นลง เช่น คลื่น 40 kHz จะมีความยาวคลื่นประมาณ 8 มม.ซึ่งเล็กกว่าช่องเปิดของคลื่น ทำให้เสียงไม่มีการหักเห หรือเลี้ยวเบน ทำให้คลื่นเสียงที่ส่งออกมาจะพุ่งออกมาเป็นลำแคบๆ ซึ่งหมายความว่า คลื่นเสียงนั้นมีทิศทางนั่นเอง ดังตัวอย่างในรูป



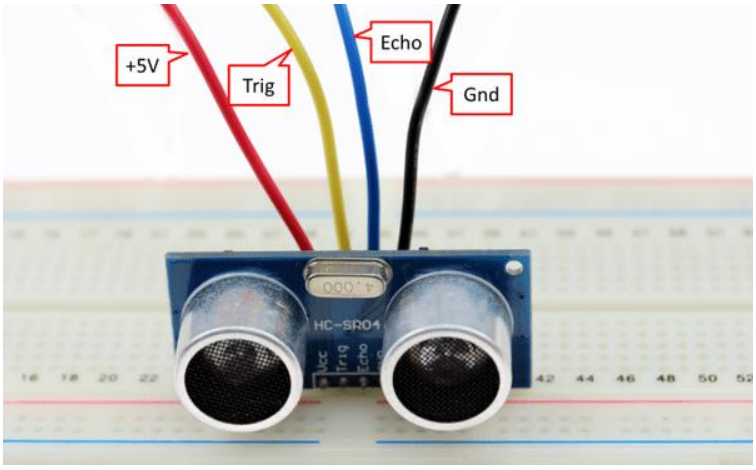
<u>รูปที่ 2.1.1 แสดงการหักเบนของคลื่นเสียงแบบกระจายตัว (รูปบน) และ คลื่นเสียงที่มีทิศทาง (รูปล่าง)</u>

เมื่อเสียงที่ส่งออกมามีทิศทางนั้น เราจึงสามารถนำคลื่นเสียงไปใช้งานได้หลายอย่าง ทั้งการวัดความลึกใต้ท้องทะเลโดยใช้โซนาร์ การจับตำแหน่งอวัยวะต่างๆในร่างกาย ใช้ทดสอบการรั่วของท่อ ซึ่งจะใช้ความถี่ที่แตกต่างกันไปตามงานที่นำไปใช้ เช่น คลื่นเสียงที่เดินทางผ่านอากาศนั้นจะต้องไม่เกิน 50 kHz เนื่องจากถ้าความสูงมากกว่านี้ อากาศจะดูดซับความแรงของคลื่นเสียงไปจนหมดนั้นเอง ส่วนในทางการแพทย์ที่ต้องการ ระยะในการตรวจจับสั้นๆ แต่แม่นยำสูง ก็จะใช้คลื่นเสียงที่มีความถี่มากๆ ซึ่งอยู่ที่ประมาณ 1 MHz ถึง 10 MHz จะเห็นได้ว่าคลื่นเสียงUltrasonic นั้น สามารถนำไปใช้กับงานรูปแบบต่างๆได้อย่างมากมาย

<u>Features : ข้อมูลสำคัญ</u>

ใช้แรงดันประมาณ :+5V DC
Quiescent Current : <2mA
ทำงานโดยใช้กระแสประมาณ: 15mA
ความกว้างเชิงมุมในการวัด: <15°
ช่วงการวัดระยะทาง : 2cm – 400 cm/1" - 13ft
ความละเอียด : 0.3 cm
Measuring Angle: 30 degree
Trigger Input Pulse width: 10uS
Dimension: 45mm x 20mm x 15mm
ตัวเซ็นเซอร์จะมีอยู่ 4 ขา
1. ขา VCC สำหรับต่อแรงดันไฟเลี้ยงไม่เกิน 5V

2. ขา Trig เป็นขาอินพุตรับสัญญาณพัลส์ความกว้าง 10 ไมโครวินาทีเพื่อกระตุ้นการสร้าง Ultrasonic ความถี่ 40kHz ออกสู่อากาศจากตัวส่ง
3. ขา Echo เป็นขาเอาต์พุตสำหรับส่งสัญญาณพัลส์ออกจากโมดูลไปยังไมโครคอนโทรลเลอร์ เพื่อตรวจจับความกว้างของสัญญาณพัลส์และคำนวณเป็นระยะทาง
4. ขา GND สำหรับต่อจุดกราวด์ร่วมแรงดัน และสัญญาณ

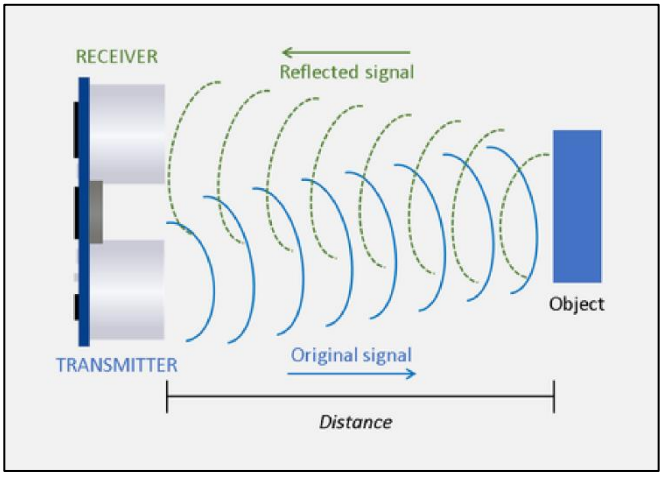


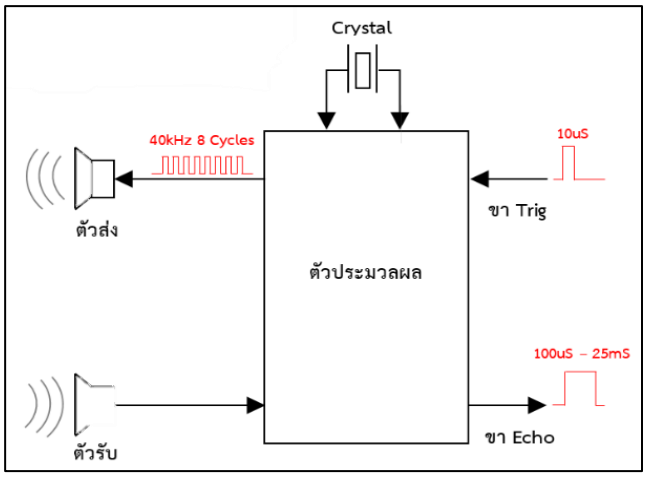
<u>รูปที่ 2.1.2 แสดงขาต่างๆของ Ultrasonic sensor HC-SR04</u>

<u>หลักการทำงาน</u>

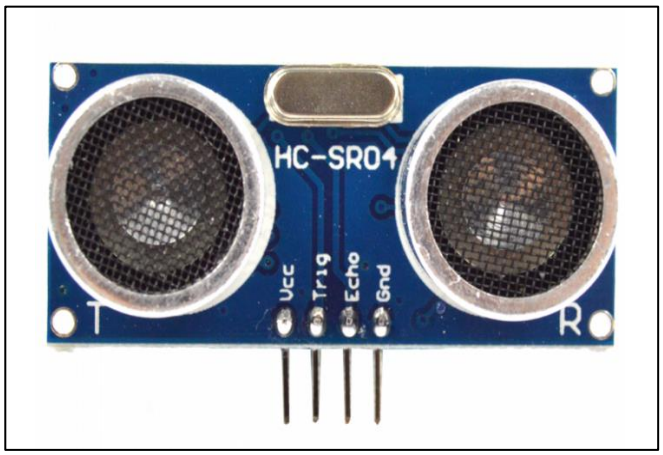
โมดูล HC-SR04 เป็นอุปกรณ์อิเล็กทรอนิกส์ สำหรับวัดระยะห่างด้วยคลื่น Ultrasonic (ใช้คลื่นเสียงความถี่ ประมาณ 40kHz) มีสองส่วนหลักคือ ตัวส่งคลื่นที่ทำหน้าที่สร้างคลื่นเสียงออกไปในการวัดระยะแต่ละครั้ง("Ping") แล้วเมื่อไปกระทบวัตถุหรือสิ่งกีดขวาง คลื่นเสียงถูกสะท้อนกลับมายังตัวรับแล้วประมวลผลด้วยวงจอิเล็กทรอนิกส์ภายในโมดูล ถ้าจับเวลาในการเดินทางของคลื่นเสียงในทิศทางไปและกลับ และถ้าทราบความเร็วเสียงในอากาศ ก็จะสามารถคำนวณระยะห่างจากวัตถุกีดขวางได้

สิ่งที่จะต้องทราบคือ Ultrasonic คือเสียง ซึ่งความเร็วในการเดินทางของเสียงจะแปรผันตามอุณหภูมิที่เปลี่ยนแปลงไป จึงอาจทำให้ค่าระยะทางที่วัดได้มีความคลาดเคลื่อน แต่ Sensor บางรุ่นจะมีเซ็นเซอร์
วัดอุณหภูมิมาด้วย ทำให้สามารถวัดระยะทางได้แม่นยำมากขึ้น สำหรับรุ่นที่ไม่มี อาจใช้เซนเซอร์วัดอุณหภูมิมาใช้เพื่อแก้ไขค่าผิดพลาดเองได้ หรือใช้อุณหภูมิเฉลี่ยของประเทศไทยที่ 27 องศาเซลเซียสก็ได้อีกส่วนที่จะต้องรู้คือช่วงวัดและมุมที่สามารถวัดได้ เช่น Sensor บางตัวอาจมีระยะที่วัดได้ระหว่าง 2 ซ.ม. - 400 ซ.ม. และมีมุมวัด 15 องศา เป็นต้น ทั้งนี้ขึ้นกับรูปร่างลักษณะของลำโพงที่ใช้กำเนิดเสียงด้วย





รูปที่ 2.1.3 และ รูปที่ 2.1.4 แสดงการทำงานของ Ultrasonic sensor HC-SR04



2.1.4 โมดูล Ultrasonic sensor HC-SR04

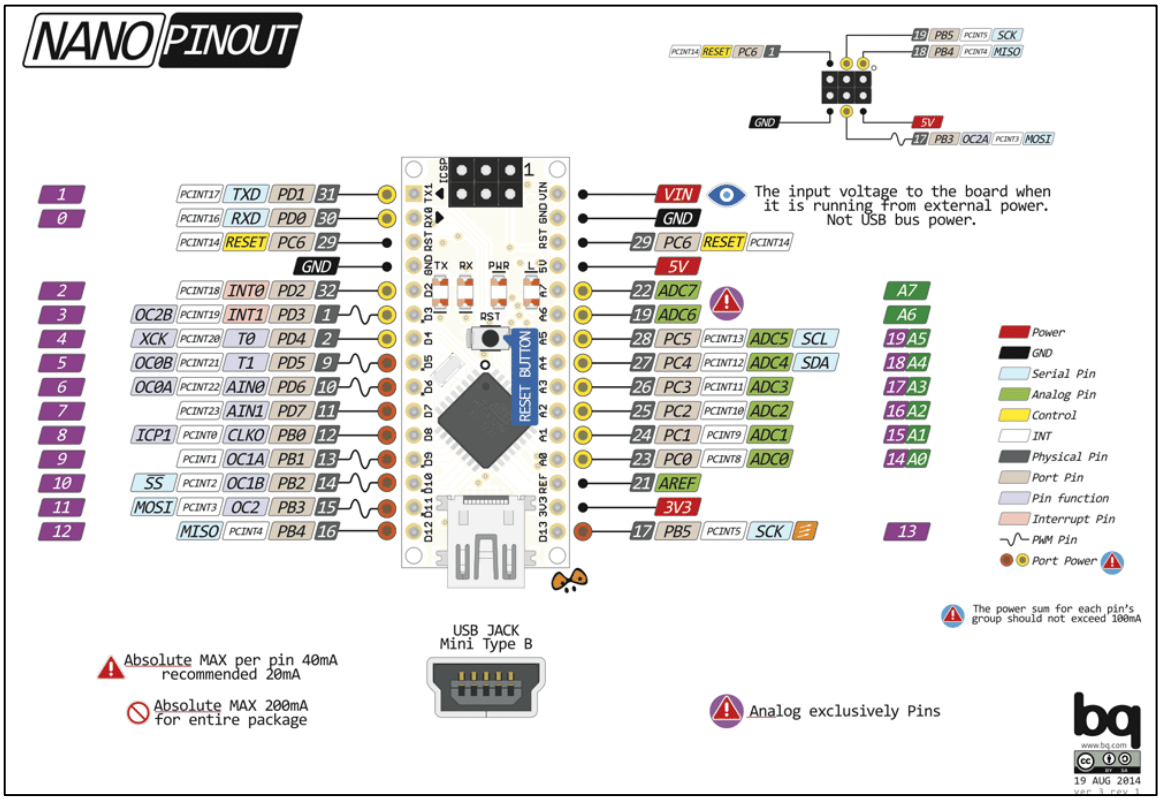
2..2 Arduino Nano v3

บอร์ด Arduino Nano 3.0 มีฟังก์ชันและความสามารถเหมือนกับบอร์ด Arduino รุ่น Uno R3 , Pro mini , Duemilanove เพราะใช้ชิฟตัวเดียวกันคือ Atmega328 โดยรุ่น nano 3.0 เปลี่ยนหัว DC Jack เป็นการต่อกับขา VIN แทน และส่วนหัวเชื่อมต่อกับเครื่องคอมเป็นแบบ Mini USB ทำให้มีขนาดเล็กที่สุดหรือสรุปง่าย ๆ บอร์ด Arduino รุ่น Nano 3.0 คือบอร์ด Arduino รุ่น Uno R3 ที่ย่อส่วนให้เล็กลงมา ประหยัดพื้นที่ ราคาถูกลง แต่ความสามารถยังเท่าเดิมเหมือนกับ Uno R3

ข้อมูลจำเพาะ Arduino Nano v3

ซอฟต์แวร์ Arduino (IDE) ใช้สำหรับโปรแกรม Arduino Nano ซอฟต์แวร์ Arduino เป็นสภาพแวดล้อมการพัฒนาแบบบูรณาการที่ใช้ร่วมกันกับบอร์ด Arduino ทั้งหมดและทำงานทั้งแบบออนไลน์และออฟไลน์

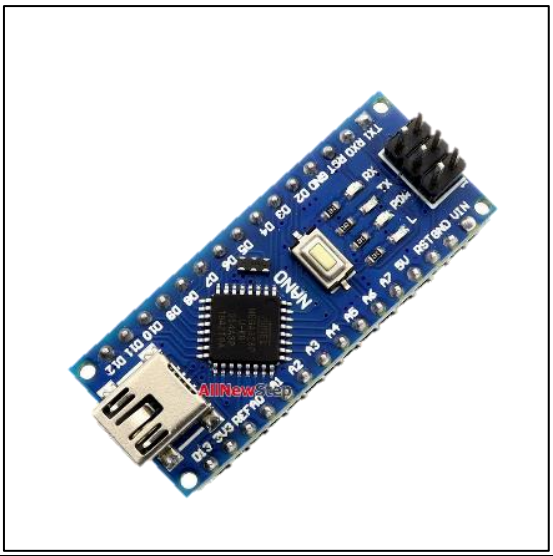
ข้อกำหนดรายละเอียดของบอร์ด Arduino Nano มีดังต่อไปนี้:
ไมโครคอนโทรลเลอร์ ATmega328
แรงดันไฟฟ้าที่ใช้งาน (ระดับตรรกะ): 5 V
แรงดันไฟฟ้าขาเข้า (แนะนำ): 7-12 V
แรงดันไฟฟ้าขาเข้า (จำกัด): 6-20 V
Digital I / O Pins: 14 (ซึ่ง 6 ขาเอาต์พุต PWM)
ช่องใส่แบบอะนาล็อก: 8
กระแสไฟกระแสตรงต่อขา I / O: 40 mA
หน่วยความจำแฟลช 32 KB (ATmega328) ซึ่งมี 2 กิโลไบต์ใช้สำหรับบูตโหลด
SRAM: 2 KB (ATmega328)
EEPROM: 1 KB (ATmega328)
ความเร็วสัญญาณนาฬิกา: 16 MHz
การวัด: 0.73 "x 1.70"



รูปที่ 2.2.1 แสดงขาต่างๆของ Arduino Nano v3

บอร์ดรุ่น Nano 3.0 ขาใช้งาน Pin Out

Pin Category	Pin Name	Details
Power	Vin, 3.3V, 5V, GND	Vin: Nano 3.0 ขาใช้งานนี้สำหรับ ไฟเข้าสำหรับจ่ายไฟให้บอร์ด Arduino โดยใช้ไฟจากภายนอก ไม่ผ่านช่อง USB รับไฟได้ช่วง 6-12V 5V: Nano 3.0 ขาใช้งานนี้คือ ขาไฟออก 5V สำหรับจ่ายให้อุปกรณ์ Arduino Sensor ต่าง ๆ 3.3V:ขาไฟออก 3.3V จ่ายกระแสได้สูงสุดเพียง 50mA GND: ขากราวน์
Reset	Reset	ขารีเซต สำหรับรีเซตบอร์ด nano 3.0
Analog Pins	A0 – A7	Nano 3.0 ขาใช้งานแบบ Analog สำหรับรับค่าแบบ Analog เป็นไฟในช่วง 0-5V
Input/Output Pins	Digital Pins D0 – D13	Nano 3.0 ขาใช้งาน แบบ Digital สามารถเป็นได้ทั้งแบบ Input และ Output 0V (low) และ 5V (high)
Serial	Rx, Tx	Nano 3.0 ขาใช้งาน สำหรับรับและส่งข้อมูล สื่อสารแบบ Serial TTL
External Interrupts	2, 3	Nano 3.0 ขานี้ใช้ฟังก์ชัน Interruptinterrupt ได้
PWM	3, 5, 6, 9, 11	Nano 3.0 ขาสำหรับเอาต์พุตแบบ PWM ความละเอียด 8 บิต หรือช่วงเลข 0-255
SPI	10 (SS), 11 (MOSI), 12 (MISO) and 13 (SCK)	ใช้สำหรับสื่อสารแบบ SPI
Inbuilt LED	13	ขานี้มี LED ที่ติดตั้งมาให้ในบอร์ด ให้เราทดลองได้รวดเร็ว
IIC	A4 (SDA), A5 (SCA)	ใช้สำหรับสื่อสารแบบ I2C



AREF	AREF	ใช้เป็นแรงดันไฟฟ้าอ้างอิง หรับไฟ input

บอร์ดทดลอง Arduino รุ่น Nano 3.0 ใช้ไฟเลี้ยง 5 โวลต์ การจ่ายไฟทำได้ 3 แบบคือ

1. Arduino บอร์ด Nano 3.0 จ่ายไฟ 5V ที่ช่อง USB ตอนที่เราเสียบ USB กับเครื่องคอม บอร์ดจะได้รับไฟ 5 โวลต์จากเครื่องคอมผ่านสาย USB ดังนั้น เราไม่ต้องต่อไฟเพิ่ม
2. Arduino บอร์ด Nano 3.0 ใช้ไฟ 5V โดยจ่ายไฟภายนอก 5V เข้าที่ขา 5V และ GND
3. Arduino บอร์ด Nano 3.0 ใช้ไฟ 6-12V โดยจ่ายไฟภายนอก 6-12V เข้าที่ขา VIN และ GND

รูปที่ 2.2.2 โมดูล Arduino Nano v3

2.3 Buzzer

Buzzer บัซเซอร์ คือ ลำโพงแบบแม่เหล็กหรือ แบบเปียโซที่มีวงจรกำเนิดความถี่ (oscillator) อยู่ภายในตัว ใช้ไฟเลี้ยง 3.3 - 5V สามารถสร้างเสียงเตือนหรือส่งสัญญาณที่เป็นรูปแบบต่างๆ ลำโพงบัซเซอร์เป็นอุปกรณ์ที่ให้กำเนิดเสียงทำหน้าที่เปลี่ยนสัญญาณไฟฟ้าให้อยู่ในรูปสัญญาณเสียง ลำโพงบัซเซอร์มีอยู่ 2 ประเภท ได้แก่

1. แบบแอคทีฟ (Active Buzzer) ลำโพงชนิดนี้มีวงจรกำเนิดความถี่อยู่ภายใน สามารถสร้างสัญญาณเสียงเตือนได้ทันทีเพียงแค่จ่ายแรงดันไฟฟ้าเข้าไป

2. แบบพาสซีฟ (Passive Buzzer) ลำโพงชนิดนี้ทำงานเหมือนลำโพงขนาดเล็ก คือ ถ้าป้อน แรงดันไฟฟ้ากระแสตรงเข้าไปไม่มีเสียงถ้าต้องการให้มีสัญญาณเสียงต้องทำการป้อนสัญญาณความถี่ เข้าไป ลำโพงชนิดนี้สามารถกำเนิดเสียงที่มีความแต่กต่างกันตามความถี่ที่ป้อนเข้ามาเราอาจจะเคยได้ยินเสียงบัซเซอร์อยู่บ่อยๆ เช่น เสียงปี๊บที่อยู่ในคอมพิวเตอร์ก็ใช้บัซเซอร์ในการส่งสัญญาณให้ทราบสถานะของคอมพิวเตอร์ให้ทราบว่า
มีปัญหาอะไร

รูปที่ 2.3.1 Buzzer

2.4 แบตเตอรี่ลิเธียมไอออน

ในช่วงทศวรรษที่ 1970 เริ่มมีแบตเตอรี่ปฐมภูมิ (แบตเตอรี่ที่ไม่สามารถประจุไฟใหม่ได้) ชนิดที่มีโลหะลิเทียมเป็นวัสดุขั้ววางจำหน่าย และได้มีการพัฒนาวัสดุอย่างต่อเนื่องจนได้แบตเตอรี่ทุติยภูมิที่มีโลหะลิเทียม เป็นขั้ว อย่างไรก็ตาม แบตเตอรี่ลิเทียมอาจเกิดเดนไดรต์ของโลหะลิเทียมขึ้นขณะใช้งาน ทำให้แบตเตอรี่เกิดการ ลัดวงจรจากภายในได้จึงไม่ปลอดภัย ทำให้มีการพัฒนาวัสดุขั้วที่มีความปลอดภัยสูงขึ้น กระทั่งได้สารประกอบ จำพวกลิเทียมเมทัลออกไซด์ (lithium metal oxide) เป็นวัสดุขั้วในแบตเตอรี่ลิเทียมไอออน ดังที่ใช้กันอยู่ทุกวันนี้ แบตเตอรี่ลิเทียมไอออนพัฒนามาตั้งแต่ช่วงต้นทศวรรษที่ 1980 โดยบริษัท Asahi Chemicals และได้วางตลาดในปี 1991 โดยบริษัท Sony ซึ่งแบตเตอรี่รุ่นแรกนี้ได้ใช้ในโทรศัพท์มือถือของ Kyocera จุดเด่นของแบตเตอรี่ลิเทียมไอออนคือความจุพลังงานและกำลังไฟฟ้าที่สูงกว่าแบตเตอรี่ตระกูลนิกเกิลและกรดตะกั่ว นอกจากนี้ ยังมีค่าศักย์ไฟฟ้าสูง มีอัตราการสูญเสียประจุระหว่างไม่ใช้งาน (self-discharge rate) ที่ต่ำไม่มีปรากฏการณ์ ความจำและมีความปลอดภัยสูงกว่าแบตเตอรี่ที่ใช้โลหะลิเทียมเป็นขั้ว

สำหรับคุณสมบัติหลักของแบตเตอรี่ลิเธียมไอออน คือ การจ่ายไฟที่แรง และคงที่อยู่ตลอดเวลา แม้ไฟในแบตเตอรี่ใกล้จะหมด แถมยังมีระยะเวลาการชาร์จไฟจนเต็มความจุที่เร็วกว่าแบตเตอรี่แบบอื่นๆ และยังใช้ได้นานกว่าอีกด้วยเช่นกัน

อายุการใช้งานของแบตเตอรี่ลิเธียมไอออน ในระยะที่เต็มประสิทธิภาพ จะอยู่ระหว่าง 1.0-1.5 ปี ขึ้นอยู่กับปริมาณการใช้งานว่ามากหรือน้อยรวมไปถึงการดูแลรักษา และหลังจากนั้นก็จะเสื่อมลง และถึงแม้ว่าเราจะเก็บแบตเตอรี่ชนิดนี้เอาไว้เฉยๆ โดยไม่ได้ใช้งานอะไรเลย แบตเตอรี่ก็สามารถเสื่อมประสิทธิภาพลงได้อยู่ดี

การดูแลรักษาแบตเตอรี่แบบลิเธียมไอออน

- เก็บแบตเตอรี่ หรืออุปกรณ์ที่ใช้แบตเตอรี่ชนิดนี้ให้อยู่ภายในอุณหภูมิ และความชื้นที่เหมาะสม หลีกเลี่ยงการชาร์จไฟภายใต้อุณหภูมิที่สูง (เพราะจะยิ่งทำให้แบตเตอรี่เสื่อมเร็วขึ้น)
- อย่าใช้งานจนแบตเตอรี่หมดเกลี้ยง หรือใกล้หมดสุดๆ ควรหมั่นชาร์จให้มีไฟเลี้ยงตัวแบตเตอรี่อยู่บ่อยๆ เพราะการชาร์จไฟบ่อยๆ ไม่ได้มีผลต่อการเสื่อมสภาพของแบตเตอรี่
- ใช้อุปกรณ์ชาร์จไฟที่ได้มาตรฐาน มีการจ่ายไฟเข้าแบตเตอรี่ที่นิ่ง และคงที่ ไม่ควรชาร์จไฟในรถบ่อยๆ เพราะไฟในรถไม่ค่อยนิ่งเท่าไหร่นัก

ประเภทของแบตเตอรี่ลิเธียมไอออน

แบตเตอรี่ลิเทียมไอออนที่มีใช้ในปัจจุบันมี 6 ประเภทหลัก โดยทั่วไปจะแบ่งตามวัสดุที่ใช้ทำขั้วบวก ส่วนขั้วลบทำจากแกรไฟต์เป็นหลักแต่จะมีประเภท LTO (Lithium Titanate) ที่แตกต่างออกไปคือมีขั้วลบเป็นลิเทียมไททาเนต ทั้งนี้เนื่องจากแบตเตอรี่ลิเทียมไอออนแต่ละประเภทมีสมบัติแตกต่างกัน จึงเหมาะสมต่อการใช้งานที่แตกต่างกันไปด้วย

ประเภทที่	วัสดุขั้วบวก	วัสดุขั้วลบ	การใช้งาน
1	Lithium Cobalt Oxide ($LiCoO_2$, LCO)	แกรไฟต์	โทรศัพท์มือถือ แท็บเล็ต แล็ปท็อป กล้องดิจิทัล
2	Lithium Manganese Oxide ($LiMn_2O_4$, LMO)	แกรไฟต์	เครื่องมือไฟฟ้า (Power tools) อุปกรณ์การแพทย์ ระบบส่งกำลังในยานพาหนะไฟฟ้า
3	Lithium Nickel Manganese Cobalt Oxide ($Li(Ni,Mn,Co)O_2$, NMC,NCM)	แกรไฟต์	จักรยานไฟฟ้า อุปกรณ์การแพทย์ ระบบส่งกำลังในยานพาหนะไฟฟ้า (มักใช้ในรถไฮบริด) ระบบสำรองไฟฟ้า
4	Lithium Nickel Cobalt Aluminum Oxide ($Li(Ni,Co,Al)O_2$, NCA)	แกรไฟต์	อุปกรณ์การแพทย์ ระบบส่งกำลังในยานพาหนะไฟฟ้า (เช่นที่พบใน Tesla Model S) ระบบสำรองไฟฟ้า
5	Lithium Iron Phosphate ($LiFePO_4$, LFP)	แกรไฟต์	ระบบส่งกำลังในยานพาหนะไฟฟ้า หรือแทนแบตเตอรี่กรดตะกั่วในรถยนต์ (Start-Lighting-Ignition battery) ระบบที่ต้องการกระแสและความทนทานสูง
6	แกรไฟต์ หรือ LMO	Lithium Titanate ($Li_4Ti_5O_{12}$, LTO)	ระบบสำรองไฟฟ้า ระบบส่งกำลังในยานพาหนะไฟฟ้า (Mitsubishi i-MiEV, Honda Fit EV)

รูปที่ 2.4.1 ประเภทของแบตเตอรี่ลิเธียมไอออน

<u>หลักการทำงาน</u>

แบตเตอรี่ลิเทียมไอออนมีส่วนประกอบที่สำคัญ คือ ขั้วลบ มีองค์ประกอบหลักเป็นคาร์บอนที่มีรูพรุน (เช่น แกรไฟต์) เคลือบบนแผ่นทองแดง ขั้วบวก เป็นลิเทียมเมทัลออกไซด์เคลือบบนแผ่นอะลูมิเนียม – สารละลายอิเล็กโทรไลต์ ประกอบด้วยเกลือของลิเทียม เช่น LiPF 6 หรือ LiBF 4 ในตัวทำละลาย เช่น เอทิลีนคาร์บอเนต (ethylene carbonate) ไดเอทิลคาร์บอเนต (diethyl carbonate) และ/หรือ ไดเมทิลคาร์บอเนต (dimethyl carbonate) เยื่อเลือกผ่าน (separator) กั้นระหว่างขั้วทั้งสอง ทำจากพอลิโพรพิลีน (polypropylene, PP) และ/หรือ พอลิเอทิลีน (polyethylene, PE) เมื่อมีการอัดประจุ (charge) ไอออน ของลิเทียม จะเคลื่อนออกจากโครงสร้างของขั้วบวก ผ่านเยื่อ เลือกผ่าน เข้าสู่ขั้วลบ เกิดเป็นสารประกอบ ของลิเทียม และ

คาร์บอน และขณะเดียวกันอิเล็กตรอนจะเคลื่อน จากขั้วบวกสู่ขั้วลบผ่านวงจรภายนอก และขณะเกิดการคายประจุ (discharge) ปฏิกิริยาจะเกิดใน ทางตรงกันข้าม ดังแสดงในภาพที่ 1 กระบวนการที่ ไอออนของลิเทียมสอดแทรกเข้าไปอยู่ในโครงสร้าง ของวัสดุขั้วบวกหรือขั้วลบ เรียกว่า lithium intercalation หรือ lithium insertion

รูปที่ 2.4.2 แบตเตอรี่ลิเธียม - ไอออน

2.5 มอเตอร์ (Motor)

2.5.1 ความหมายและชนิดของมอเตอร์

มอเตอร์ เป็นอุปกรณ์ไฟฟ้าที่เปลี่ยนพลังงานไฟฟ้าเป็นพลังกล มอเตอร์ที่ใช้งานในปัจจุบันแต่ละชนิดก็จะมีคุณสมบัติที่แตกต่างออกไปต้องการความเร็ว รอบหรือกำลังงานที่แตกต่างกัน ซึ่งมอเตอร์แต่ละชนิด จะแบ่งได้เป็น 2 ชนิดตามลักษณะการใช้งานกระแสไฟฟ้า

มอเตอร์ไฟฟ้าแบ่งออกตามการใช้ของกระแสไฟฟ้าได้ 2 ชนิดดังนี้

2.5.1.1 มอเตอร์ไฟฟ้ากระแสสลับ(Alternating Current Motor)หรือเรียกว่าเอ.ซี มอเตอร์(A.C. MOTOR)การแบ่งชนิดของมอเตอร์ไฟฟ้าสลับแบ่งออกเป็น 3 ชนิดได้แก่

1. มอเตอร์ไฟฟ้ากระแสสลับชนิด 1 เฟส หรือเรียกว่าซิงเกิลเฟสมอเตอร์ (A.C. Sing Phase) จะใช้กับแรงดันไฟฟ้า 220 โวลต์มีสายไฟ เข้า 2 สาย มีแรงม้าไม่สูง ส่วนใหญ่ตามบ้านเรือน

- สปลิทเฟส มอเตอร์(Split-Phase motor)
- คาปาซิเตอร์มอเตอร์ (Capacitor motor)
- รีพัลชั่นมอเตอร์ (Repulsion-type motor)
- ยูนิเวอร์แวซลมอเตอร์ (Universal motor)
- เช็ดเดดโพล มอเตอร์ (Shaded-pole motor)

2. มอเตอร์ไฟฟ้าสลับชนิด 2 เฟสหรือเรียกว่าทูเฟสมอเตอร์ (A.C.Two phas Motor)

3. มอเตอร์ไฟฟ้ากระแสสลับชนิด 3 เฟสหรือเรียกว่าทีเฟสมอเตอร์ (A.C. Three phase Motor) เป็นมอเตอร์ที่ใช้ในงานอุตสาหกรรมต้องใช้ระบบไฟฟ้า 3 เฟส ใช้แรงดัน 380 โวลต์มีสายไฟเข้ามอเตอร์ 3 สาย

2.5.1.2 มอเตอร์ไฟฟ้ากระแสตรง (Direct Current Motor)หรือเรียกว่าดี.ซี มอเตอร์(D.C. MOTOR)การแบ่งชนิดของมอเตอร์ไฟฟ้ากระแสตรงแบ่งออกได้ดังนี้

มอเตอร์ไฟฟ้ากระแสตรงแบ่งออกเป็น 3 ชนิดได้แก่

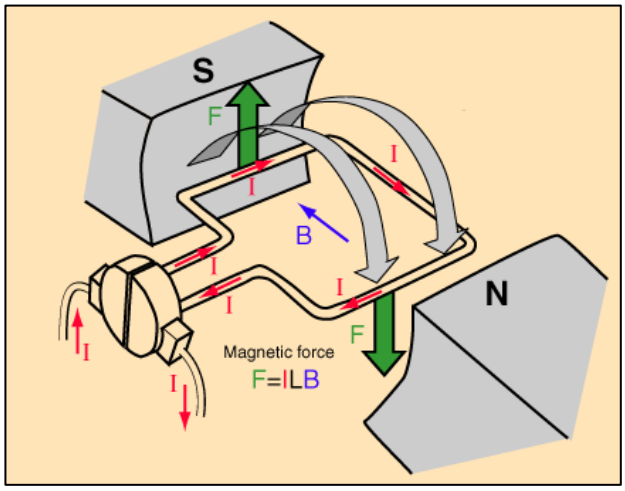
1.มอเตอร์แบบอนุกรมหรือเรียกว่าซีรีส์มอเตอร์ (Series Motor)

2.มอเตอร์แบบอนุขนานหรือเรียกว่าชันท์มอเตอร์ (Shunt Motor)

3.มอเตอร์ไฟฟ้าแบบผสมหรือเรียกว่าคอมเปาวด์มอเตอร์ (Compound Motor)

<u>การทำงานของมอเตอร์</u>

การทำงานเบื้องต้นของมอเตอร์ไฟฟ้ากระแสตรง มีแรงดันไฟตรงจ่ายผ่านแปรงถ่านไปคอมมิวเตเตอร์ ผ่านไปให้ขดลวดตัวนำที่อาร์เมเจอร์ ทำให้ขดลวดอาร์เมเจอร์เกิดสนามแม่เหล็กไฟฟ้าขึ้นมา ทางด้านซ้ายมือเป็นขั้วเหนือ (N) และด้านขวาเป็นขั้วใต้ (S) เหมือนกับขั้วแม่เหล็กถาวรที่วางอยู่ใกล้ๆ เกิดอำนาจแม่เหล็กผลักดันกันอาร์เมเจอร์หมุนไปในทิศทางตามเข็มนาฬิกาพร้อมกับคอมมิวเตเตอร์หมุนตามไปด้วย แปรงถ่านสัมผัสกับส่วนของคอมมิวเตเตอร์ เปลี่ยนไปในอีกปลายหนึ่งของขดลวด แต่มีผลทำให้เกิดขั้วแม่เหล็กที่อาร์เมเจอร์เหมือนกับขั้วแม่เหล็กถาวรที่อยู่ใกล้ๆอีกครั้ง ทำให้อาร์เมเจอร์ยังคงถูกผลักให้หมุนไปในทิศทางตามเข็มนาฬิกาตลอดเวลา เกิดการหมุนของอาร์เมเจอร์คือมอเตอร์ไฟฟ้าทำงาน



รูปที่ 2.5.1 การทำงานของมอเตอร์

2.5.2 มอเตอร์สั่น (Motor Vibrator)

มีหลักการทำงานและโครงสร้างภายในเหมือนกับมอเตอร์ไฟฟ้าทั่วไป เพียงแต่เพิ่มส่วนปลายของมอเตอร์จะมีเหล็กเป็นตัวเหวี่ยงเข้ามาทำให้เกิดการสั่นหรือการเขย่าเกิดขึ้น

รูปที่ 2.5.2 มอเตอร์สั่น (Motor Vibrator)

2.6 โปรแกรม Arduino IDE โครงสร้างของการเขียนโปรแกรม Arduino

โครงสร้างของการเขียนโปรแกรมแบ่งออกเป็นส่วนย่อย หลายๆส่วน โดยเรียกแต่ละส่วนว่า ฟังก์ชัน และ เมื่อนำฟังก์ชันมารวมเข้าด้วยกันก็จะเรียกว่าโปรแกรม โดยโครงสร้างการเขียนโปรแกรมของ Arduino นั้น ทุกโปรแกรมจะต้องประกอบไปด้วยฟังก์ชันจำนวนเท่าใดก็ได้แต่อย่างน้อยที่สุด ต้องมีฟังก์ชัน จำนวน 2 ฟังก์ชัน คือ setup() และ loop() โครงสร้างพื้นฐานของภาษาซีที่ใช้กับ Arduino นั้นจะประกอบไปด้วย 3 ส่วนคือ

2.6.1 Header ในส่วนนี้จะมีหรือไม่มีก็ได้ ถ้ามีต้องกำหนดไว้ในส่วนเริ่มต้นของโปรแกรม

2.6.2 setup() ในส่วนนี้เป็นฟังก์ชันบังคับที่ต้องกำหนดให้มีในทุกๆโปรแกรม ถึงแม้ว่าในบางโปรแกรม
จะไม่ต้องการใช้งานก็ยังจำเป็นต้องประกาศไว้ด้วยเสมอ เพียงแต่ไม่ต้องเขียนคำสั่งใดๆไว้ในระหว่างวงเล็บปีกกา {} ที่ใช้เป็นตัวกำหนดขอบเขตของฟังก์ชัน โดยฟังก์ชันนี้จะใช้สำหรับบรรจุคำสั่งในส่วนที่ต้องการให้โปรแกรมทำงานเพียงรอบเดียวตอนเริ่มต้นทำงานของโปรแกรมครั้งแรกเท่านั้น ซึ่งได้แก่คำสั่งเกี่ยวกับการ Setup ค่าการทำงานต่างๆ เช่น การกำหนดหน้าที่การใช้งานของ PinMode และการกำหนดค่า Baudrat สำหรับใช้งานพอร์ตสื่อสารอนุกรม เป็นต้น

3 loop() เป็นส่วนฟังก์ชันบังคับที่ต้องกำหนดให้มีในทุกๆโปรแกรมเช่นเดียวกันกับฟังก์ชัน setup() โดยฟังก์ชัน loop()นี้ จะใช้บรรจุคำสั่งที่ต้องการให้โปรแกรมทำงาน เป็นวงรอบซ้ำๆกันไปไม่รู้จบ ซึ่งถ้าเปรียบเทียบกับรูปแบบของ ANSI-C ส่วนนี้ก็คือ ฟังก์ชัน main() นั่นเอง

หน้าที่ของฟังก์ชันของ Arduino

#include <header.h>

เมื่อพบคำสั่ง #include ตัวแปลภาษาของ Arduino จะไปค้นหาไฟล์ที่ระบุไว้ในเครื่องหมาย <> หลังคำสั่ง #include จากตำแหน่ง Directory ที่เก็บไฟล์ Library ของโปรแกรม Arduino ไว้ ซึ่ง แน่นอนว่าส่วนของ Header จะนับรวมไปถึง คำสั่งส่วนที่ใช้ประกาศสร้าง ตัวแปร (Variable Declaration) และค่าคงที่ (Constant Declaration) รวมทั้ง ฟังก์ชันต่างๆ (Function Declaration) ด้วย ซึ่งจากตัวอย่างได้แก่ส่วนที่เป็นคำสั่ง สำหรับส่วนที่สำคัญที่สุดและขาดไม่ได้ คือ ฟังก์ชัน setup() และ ฟังก์ชัน Loop) ซึ่งฟังก์ชัน ทั้ง 2 ส่วนนี้ มีรูปแบบโครงสร้างที่เหมือนกัน แต่ถูกกำหนดด้วยชื่อ ฟังก์ชันเป็นการเฉพาะ คือ setup() และ loop() โดย setup() จะเขียนไว้ก่อน loop() ซึ่งทั้ง 2 ฟังก์ชัน นี้ มีขอบเขต เริ่มต้นและสิ้นสุด อยู่ภายใต้เครื่องหมาย{} Voidsetup()

{

คำสั่งต่างๆ ที่ต้องการเขียนไว้ภายใต้ฟังก์ชัน setup()

}

หน้าที่ของฟังก์ชัน setup() ใน Arduino คือ ใช้ทำหน้าที่เป็นส่วนของโปรแกรมย่อยสำหรับใช้ บรรจุคำสั่งต่างๆที่ใช้สำหรับกำหนดการทำงานของระบบ หรือ กำหนดคุณสมบัติการทำงานให้กับ อุปกรณ์ต่างๆซึ่งคำสั่งทั้งหมดที่บรรจุไว้ภายใต้ฟังก์ชันของ Setup() นี้ จะถูกเรียกขึ้นมาทำงานเพียงรอบเดียวคือตอนเริ่มต้นการทำงานของโปรแกรม โดยคำสั่งที่นิยมบรรจุไว้ในฟังก์ชันส่วนนี้ ได้แก่ คำสั่ง สำหรับกำหนดโหมดการทำงานของ Digital Pin หรือ คำสั่งสำหรับ กำหนดคุณสมบัติของพอร์ตสื่อสาร อนุกรม เป็นต้น

voidloop()

{

คำสั่งต่างๆที่ต้องการให้ทำงานภายใต้ฟังก์ชัน loop()

}

หน้าที่ของฟังก์ชัน loop() ใน Arduino คือใช้ทำหน้าที่เป็นส่วนของโปรแกรมหลัก สำหรับใช้ บรรจุคำสั่งควบคุมการทำงานต่างๆของโปรแกรม ที่ต้องการใช้โปรแกรมทำงาน โดยคำสั่งที่บรรจุไว้ใน ฟังก์ชันนี้จะถูกเรียกขึ้นมาทำงานซ้ำๆ กันตามลำดับและเงื่อนไขที่กำหนดไว้

ภาษา C++

ภาษา C++ เป็นภาษาโปรแกรมคอมพิวเตอร์อเนกประสงค์มีโครงสร้างภาษาที่มีการจัดชนิด ข้อมูลแบบสแตติก (Statically typed) และสนับสนุนรูปแบบการเขียนโปรแกรมที่หลากหลาย (multiparadigm language) ได้แก่ การโปรแกรมเชิงกระบวนคำสั่ง การนิยามข้อมูล การโปรแกรมเชิงวัตถุ และการโปรแกรมแบบเจเนริก (generic programming) ภาษา C++ เป็นภาษาโปรแกรมเชิง พาณิชย์ที่นิยมมากภาษาหนึ่งนับตั้งแต่ช่วงทศวรรษ 1990 เบียเนอ สเดราสดร็อบ (Bjarne Stroustrap) จากเบลล์แล็บส์ (BellLabs) เป็นผู้พัฒนาภาษา C++ (เดิมใช้ชื่อ "C with classes") ในปี ค.ศ. 1983 เพื่อพัฒนาภาษาซีดั้งเดิม สิ่งที่พัฒนาขึ้นเพิ่มเติมนั้นเริ่มจากการเพิ่มเติมการสร้างคลาสจากนั้นก็เพิ่ม คุณสมบัติต่างๆตามมา ได้แก่ เวอร์ชวลฟังก์ชันการโอเวอร์โหลดโอเปอเรเตอร์การสืบทอดหลายสาย เทมเพลต และการจัดการเอกเซพชัน มาตรฐานของภาษา C++ ได้รับการรับรองในปี ค.ศ. 1998 เป็น มาตรฐาน ISO/IEC 14882:1998 เวอร์ชั่นล่าสุดคือเวอร์ชันในปี ค.ศ. 2003 ซึ่งเป็นมาตรฐาน ISO/IEC 14882:2003 ในปัจจุบันมาตรฐานของภาษาในเวอร์ชั่นใหม่ (รู้จักกันในชื่อC++Ox) กำลังอยู่ในขั้น พัฒนา

รูปแบบของการออกแบบภาษา C++

ภาษา C++ ได้ถูกออกแบบมาเพื่อเป็นภาษาสำหรับการเขียนโปรแกรมทั่วไป สามารถรองรับการเขียนโปรแกรมในระดับภาษาเครื่องได้เช่นเดียวกับภาษาซี ในทางทฤษฎีภาษา C++ ควรจะมีความเร็วเทียบเท่าภาษา C แต่ในการเขียนโปรแกรมจริงนั้น ภาษา C++ เป็นภาษาที่มีการเปิดกว้างให้ โปรแกรมเมอร์เลือกรูปแบบการเขียนโปรแกรม ซึ่งทำให้มีแนวโน้มที่โปรแกรมเมอร์อาจใช้รูปแบบที่ไม่เหมาะสม ทำให้โปรแกรมที่เขียนมีประสิทธิภาพต่ำกว่าที่ควรจะเป็น และภาษา C++ นั้นเป็นภาษาที่มีความซับซ้อนมากกว่าภาษาซี จึงทำให้มีโอกาสเกิดบักขณะคอมไพล์มากกว่า ภาษา C++ ได้รับการออกแบบเพื่อเข้ากันได้กับภาษา C ในเกือบทุกกรณี มาตรฐานของภาษา C++ ถูกออกแบบมาเพื่อไม่ให้มีการเจาะจงแพลตฟอร์มคอมพิวเตอร์ ภาษา C++ ถูกออกแบบมาให้รองรับรูปแบบการเขียนโปรแกรมที่ หลากหลาย (multiparadigm)

รูปที่ 2.6 โปรแกรม Arduino IDE

2.7 โปรแกรมควบคุมการทำงาน

```
// defines pins numbers
const int trigPin1 = 5;
const int echoPin1 = 6;
const int trigPin2 = 7;
const int echoPin2 = 8;
const int trigPin3 = 9;
const int echoPin3 = 10;
const int motor = 13;
const int PIEZO = 2;
// defines variables
long Duration1;
int Distance1;
long Duration2;
int Distance2;
long Duration3;
int Distance3;
int safetyDistance;
void setup()
{
pinMode(trigPin1, OUTPUT); // Sets the trigPin as an Output
pinMode(echoPin1, INPUT); // Sets the echoPin as an Input
pinMode(trigPin2, OUTPUT); // Sets the trigPin as an Output
pinMode(echoPin2, INPUT); // Sets the echoPin as an Input
pinMode(trigPin3, OUTPUT); // Sets the trigPin as an Output
```

```
pinMode(echoPin3, INPUT); // Sets the echoPin as an Input
pinMode(motor, OUTPUT);
pinMode(PIEZO, OUTPUT);
Serial.begin(9600); // Starts the serial communication
}
void Activate1()
 {
  digitalWrite(trigPin1, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin1, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin1, LOW);
}
void Activate2()
 {
  digitalWrite(trigPin2, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin2, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin2, LOW);
}
void Activate3()
{
  digitalWrite(trigPin3, LOW);
  delayMicroseconds(2);
```

```
  digitalWrite(trigPin3, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin3, LOW);
}
void loop()
{
  Activate1();
  Duration1 = pulseIn(echoPin1, HIGH);
  Distance1 = Duration1 * 0.017;
  Activate2();
  Duration2 = pulseIn(echoPin2, HIGH);
  Distance2 = Duration2 * 0.017;
  Activate3();
  Duration3 = pulseIn(echoPin3, HIGH);
  Distance3 = Duration3 * 0.017;
  Serial.print(Distance1);
  Serial.print("       ");
  Serial.println(Distance2);
  Serial.println(Distance3);
 if (Distance1 <= 50 ) {
    digitalWrite(motor, HIGH);
    tone(PIEZO, 3000,100 );
    delay(200);
    tone(PIEZO, 3000,100 );
  }
```

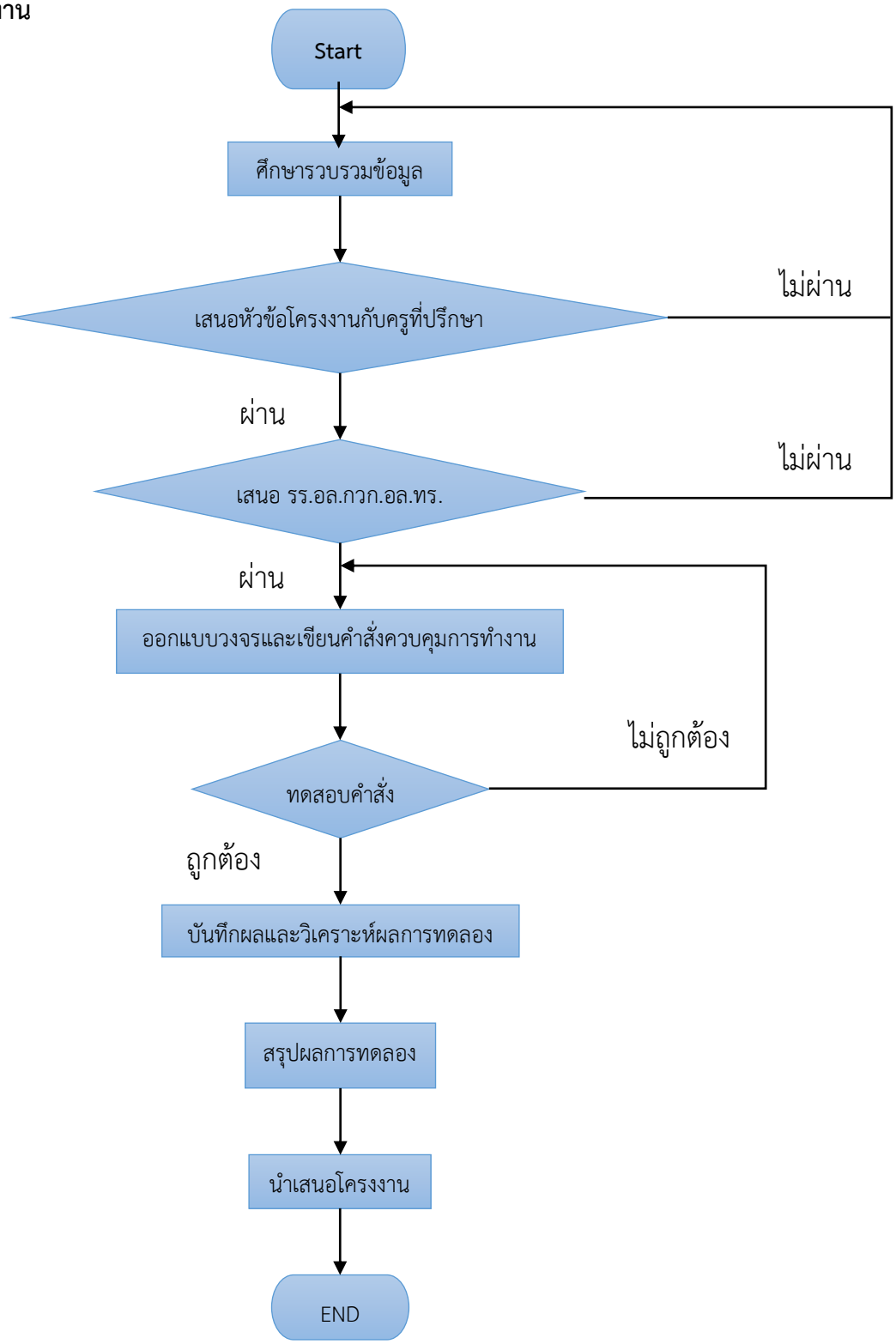
```
  Else
 {
    digitalWrite(motor, LOW);
    noTone;
  }
  if (Distance2 <= 50 )
 {
    digitalWrite(motor, HIGH);
    tone(PIEZO, 1000,100);
    delay(200);
    tone(PIEZO, 1000,300);
  }
  else
{
    digitalWrite(motor, LOW);
    noTone;
  }
  if (Distance3 <= 50 )
 {
    digitalWrite(motor, HIGH);
    tone(PIEZO, 200,100);
    delay(200);
    tone(PIEZO, 200,300);
  }
  Else
```

```
{
   digitalWrite(motor, LOW);
   noTone;
 }
 }
```

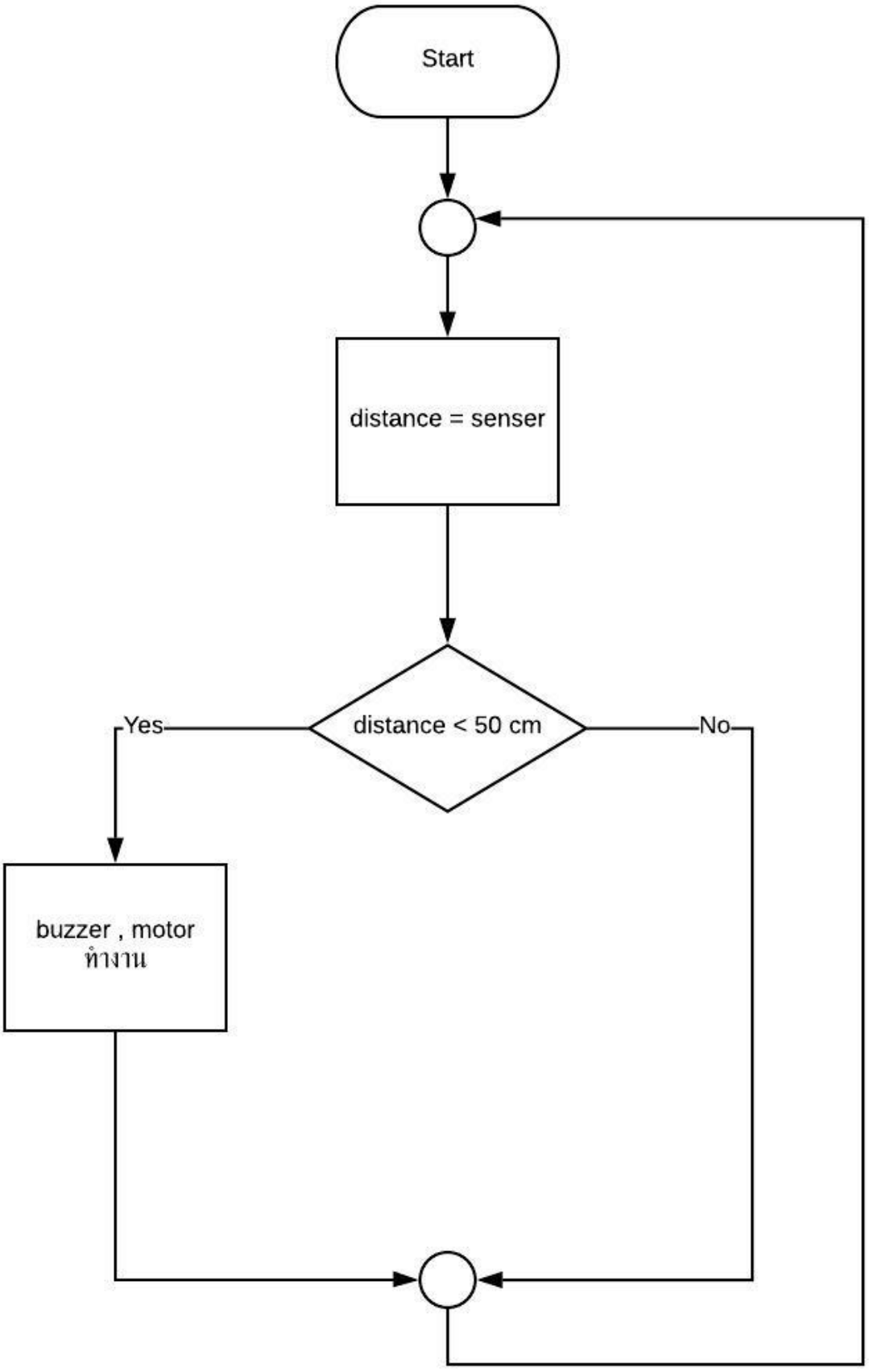
บทที่ 3

วิธีดำเนินงาน

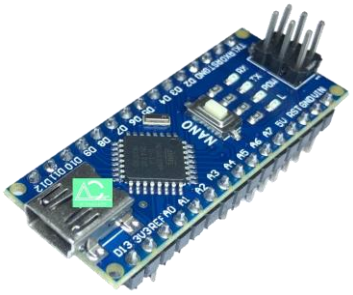
3.1 วิธีการดำเนินงาน



3.2 การทำงานของระบบ



3.3 วัสดุอุปกรณ์

ลำดับ	รายการ	จำนวน	ราคา (บาท)	ราคา (รวม)	รูปภาพประกอบ
1	Ultrasonic Sensor	3	40	120	
2	Arduino Nano V3	1	125	125	
3	Switch	1	10	10	
4	rumble motor	1	40	40	
5	กล่อง	1	50	50	

6	ไม้นำทาง	1	80	80	
7	แบตลิเธียม 3.7โวลตUltrafire	3	30	90	
8	Buzzer	1	19		

3.4 ขั้นตอนการดำเนินงาน

1 ประชุมวางแผนเรื่องหัวข้อในการทำงาน

2 แบ่งหน้าที่การทำงานให้แก่สมาชิกในกลุ่ม

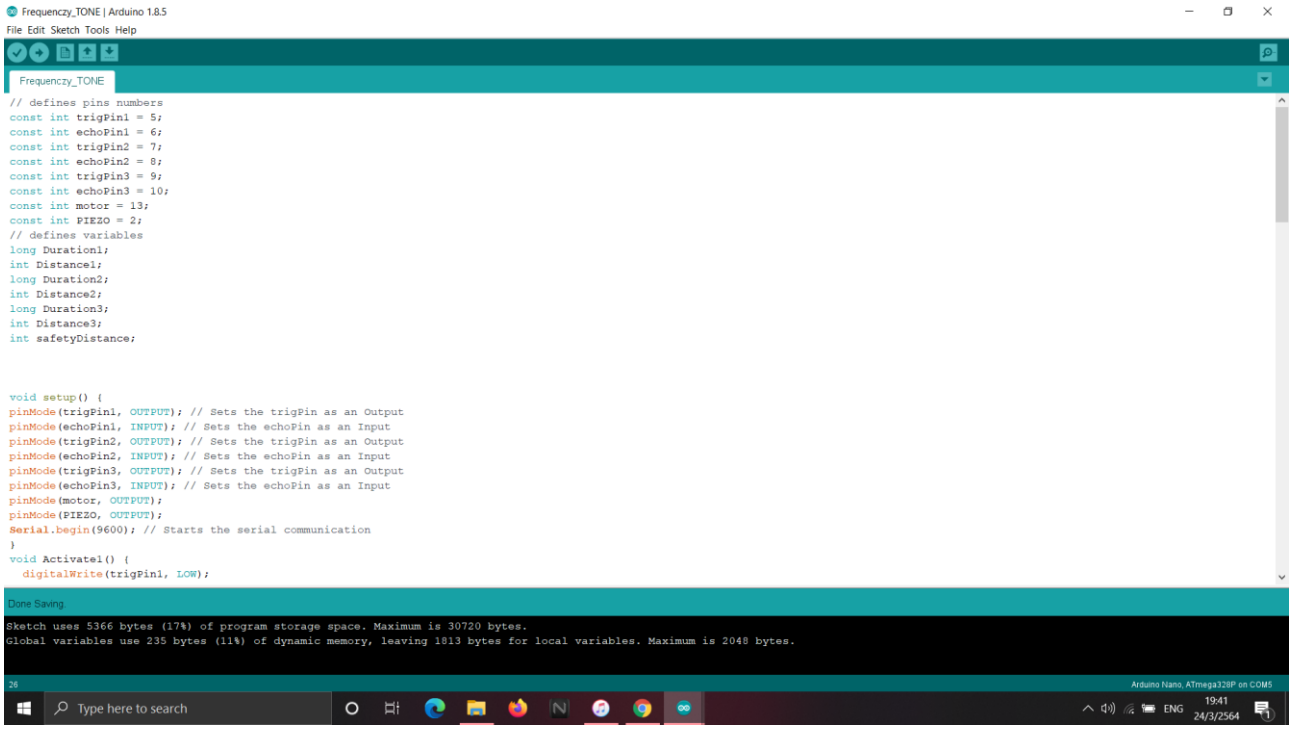
3 ค้นหาข้อมูลในการทำโครงงานเกี่ยวกับอุปกรณ์หลักการทำงานราคาของอุปกรณ์

4 ศึกษาการใช้งานโปรแกรม Arduino IDE

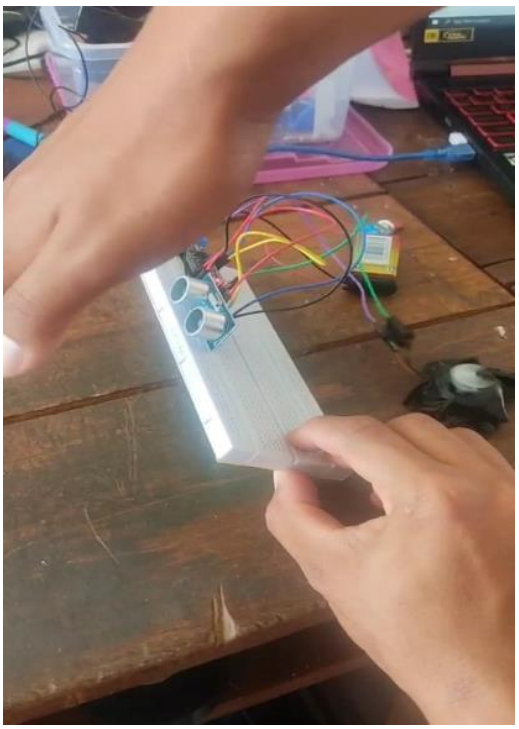
5 ซื้ออุปกรณ์ในการทำงาน

6 ทำตามหน้าที่ที่ได้รับมอบหมาย

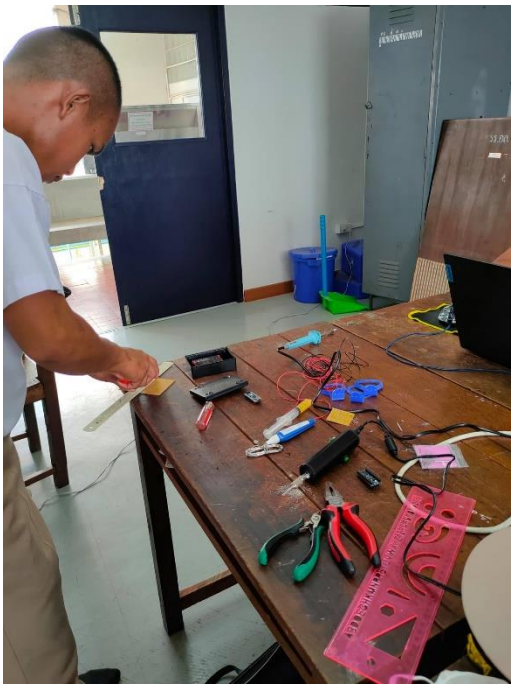
- **เขียนโปรแกรมคำสั่งการทำงาน**



- ทดลองการทำงานของโปรแกรมคำสั่ง



- ประกอบวงจรเข้ากับกล่อง



- **ประกอบกล่องติดกับไม้นำทาง**

ไม้นำทางแจ้งเตือนสิ่งกีดขวาง

บทที่ 4
ผลการดำเนินงาน

สำหรับโครงงานนี้มีผลการดำเนินงานโดยแบ่งการทำงานออกเป็น 2 ส่วนคือ

1. ตรวจจับสิ่งกีดขวาง
2. ส่วนแสดงผลการแจ้งเตือน

วัตถุ	ความสามารถในการตรวจจับ
กล่อง	ปกติ
กำแพง	ปกติ
ต้นไม้	ปกติ
รองเท้า	ปกติ

ตารางแสดงความสามารถในการตรวจจับวัตถุของไม้นำทางแจ้งเตือนสิ่งกีดขวาง

ระยะทาง (cm)	ความสามารถในการตรวจจับ
10	ตรวจจับได้
20	ตรวจจับได้
30	ตรวจจับได้
40	ตรวจจับได้
50	ตรวจจับได้
55	ตรวจจับได้แต่ไม่เสถียร
60	ตรวจจับไม่ได้

ตารางแสดงระยะที่ไม้นำทางแจ้งเตือนสิ่งกีดขวางสามารถตรวจจับได้

- ไม้นำทางแจ้งเตือนสิ่งกีดขวางสามารถใช้งานได้นานประมาณ 7 - 8 ชั่วโมง
- ใช้เวลาในการชาร์จแบตเตอรี่ประมาณ 2 ชั่วโมง 30 นาที

บทที่ 5

สรุปผลการทดลองและข้อเสนอแนะ

5.1 สรุปผลการทดลอง

โครงงานนี้เป็นการออกแบบและสร้างไม้นำทางแจ้งเตือนสิ่งกีดขวาง โดยมีการประยุกต์ใช้บอร์ด Arduino nano v3 ในการควบคุมการทำงานมีการใช้อัลตร้าโซนิคเซนเซอร์ในการตรวจจับวัตถุสิ่งกีดขวางโดยแสดงผลและแจ้งเตือนผ่าน Buzzer และMotor Vibrator ไม้นำทางแจ้งเตือนสิ่งกีดขวางสามารถใช้งานได้นานประมาณ 7 - 8 ชั่วโมง สามารถตรวจจับวัตถุได้ 3 ทิศทางที่แม่นยำ มีระยะการตรวจจับอยู่ที่ 0 - 50 เซนติเมตร ใช้เวลาในการชาร์จแบตเตอรี่ประมาณ 2 ชั่วโมง 30 นาที

5.2 ประโยชน์ที่ได้รับ

1. อำนวยความสะดวกแก่ผู้พิการทางสายตาได้โดยการเตือนว่าจะชนสิ่งกีดขวาง
2. นำความรู้ที่ได้มาใช้ในการประกอบวิชาชีพ
3. นำความรู้ทางด้านทฤษฎีมาประยุกต์ใช้ในทางปฏิบัติ
4. เป็นแนวทางในการนำเอาไปพัฒนาต่อยอดทางด้านอุตสาหกรรมด้านพาณิชย์

5.3 ข้อเสนอแนะ

1.ควรพัฒนาวิธีการชาร์จแบตเตอรี่ให้มีวิธีที่ง่ายยิ่งขึ้น
2.ควรนำ โมดูลJQ-6500หรือโมดูลเล่นเสียงมาใช้เพื่อเพิ่มประสิทธิภาพของไม้นำทางแจ้งเตือนสิ่งกีดขวาง
3.ควรติดGPSเพื่อให้ญาติสามารถทราบตำแหน่งของผู้ใช้งานไม้นำทางแจ้งเตือนสิ่งกีด

บรรณานุกรม

- https://robotsiam.blogspot.com/2016/09/ultrasonic-module-hc-sr04.html
- https://www.myarduino.net/article/66/สอนใช้งาน-เซนเซอร์-วัดระยะทาง-hc-sr04-hc--sr05-us-100-us-016
- http://th.sz-kuongshun.com/info/introduction-to-arduino-nano-v3-0-specificatio-28909659.html
- https://sites.google.com/site/mikhorkhxnthorllexr1/la-pho-ngbas-sexr-buzzer
- https://www.nissan.co.th/experience-nissan/Nissan-EV/lithium-ion-battery.html
- https://www2.mtec.or.th/th/e-magazine/admin/upload/298_63.pdf
- http://www.psptech.co.th/มอเตอร์motorคืออะไร-19171.page
- https://th.wikipedia.org/wiki/มอเตอร์
- https://www.myarduino.net/article/5/การเขียนโปรแกรมเบื้องต้นกับ-arduino-c-โครงสร้างโปรแกรมของ-arduino
- https://medium.com/iamgoangle/arduino-วัดระยะทางด้วย-ultrasonic-aad3c38cb626

ภาคผนวก

ภาคผนวก ก.

```
// defines pins numbers
const int trigPin1 = 5;
const int echoPin1 = 6;
const int trigPin2 = 7;
const int echoPin2 = 8;
const int trigPin3 = 9;
const int echoPin3 = 10;
const int motor = 13;
const int PIEZO = 2;
// defines variables
long Duration1;
int Distance1;
long Duration2;
int Distance2;
long Duration3;
int Distance3;
int safetyDistance;
void setup()
{
pinMode(trigPin1, OUTPUT); // Sets the trigPin as an Output
pinMode(echoPin1, INPUT); // Sets the echoPin as an Input
pinMode(trigPin2, OUTPUT); // Sets the trigPin as an Output
pinMode(echoPin2, INPUT); // Sets the echoPin as an Input
pinMode(trigPin3, OUTPUT); // Sets the trigPin as an Output
pinMode(echoPin3, INPUT); // Sets the echoPin as an Input
pinMode(motor, OUTPUT);
pinMode(PIEZO, OUTPUT);
Serial.begin(9600); // Starts the serial communication
}
void Activate1()
{
  digitalWrite(trigPin1, LOW);
```

```
  delayMicroseconds(2);
  digitalWrite(trigPin1, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin1, LOW);
}
void Activate2()
{
  digitalWrite(trigPin2, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin2, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin2, LOW);
}
void Activate3()
{
  digitalWrite(trigPin3, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin3, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin3, LOW);
}

void loop()
{
  Activate1();
  Duration1 = pulseIn(echoPin1, HIGH);
  Distance1 = Duration1 * 0.017;
  Activate2();
  Duration2 = pulseIn(echoPin2, HIGH);
  Distance2 = Duration2 * 0.017;
  Activate3();
  Duration3 = pulseIn(echoPin3, HIGH);
  Distance3 = Duration3 * 0.017;

  Serial.print(Distance1);
```

```
 Serial.print("        ");
 Serial.println(Distance2);
 Serial.println(Distance3);
if (Distance1 <= 50 )
{
   digitalWrite(motor, HIGH);
   tone(PIEZO, 3000,100 );
   delay(200);
   tone(PIEZO, 3000,100 );
 }
 Else
{
   digitalWrite(motor, LOW);
   noTone;
 }
 if (Distance2 <= 50 )
{
   digitalWrite(motor, HIGH);
   tone(PIEZO, 1000,100);
   delay(200);
   tone(PIEZO, 1000,300);
 }
 Else
{
   digitalWrite(motor, LOW);
   noTone;
 }
 if (Distance3 <= 50 )
{
   digitalWrite(motor, HIGH);
   tone(PIEZO, 200,100);
   delay(200);
   tone(PIEZO, 200,300);
 }
 else
```

```
{
    digitalWrite(motor, LOW);
    noTone;
 }
 }
```

ภาคผนวก ข.

- **เขียนโปรแกรมคำสั่งการทำงาน**

Frequenczy_TONE | Arduino 1.8.5

File Edit Sketch Tools Help

Frequenczy_TONE

```
// defines pins numbers
const int trigPin1 = 5;
const int echoPin1 = 6;
const int trigPin2 = 7;
const int echoPin2 = 8;
const int trigPin3 = 9;
const int echoPin3 = 10;
const int motor = 13;
const int PIEZO = 2;
// defines variables
long Duration1;
int Distance1;
long Duration2;
int Distance2;
long Duration3;
int Distance3;
int safetyDistance;


void setup() {
pinMode(trigPin1, OUTPUT); // Sets the trigPin as an Output
pinMode(echoPin1, INPUT); // Sets the echoPin as an Input
pinMode(trigPin2, OUTPUT); // Sets the trigPin as an Output
pinMode(echoPin2, INPUT); // Sets the echoPin as an Input
pinMode(trigPin3, OUTPUT); // Sets the trigPin as an Output
pinMode(echoPin3, INPUT); // Sets the echoPin as an Input
pinMode(motor, OUTPUT);
pinMode(PIEZO, OUTPUT);
Serial.begin(9600); // Starts the serial communication
}
void Activate1() {
  digitalWrite(trigPin1, LOW);
```

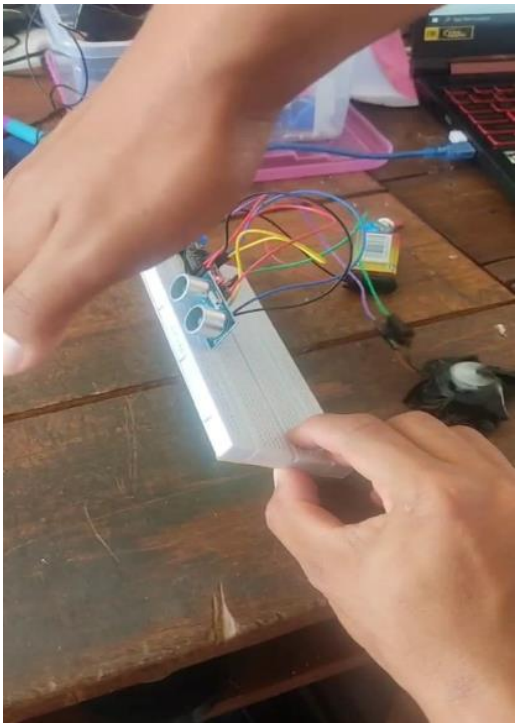
Done Saving.

```
Sketch uses 5366 bytes (17%) of program storage space. Maximum is 30720 bytes.
Global variables use 235 bytes (11%) of dynamic memory, leaving 1813 bytes for local variables. Maximum is 2048 bytes.
```

26 Arduino Nano, ATmega328P on COM5

Type here to search ENG 19:41 24/3/2564

- ทดลองการทำงานของโปรแกรมคำสั่ง



- ประกอบวงจรเข้ากับกล่อง

- ประกอบกล่องติดกับไม้นำทาง

- ไม้นำทางแจ้งเตือนสิ่งกีดขวาง

ภาคผนวก ค.

คู่มือการใช้งาน

1. เปิดสวิตช์ที่กล่องควบคุม

2. ใช้ไม้นำทางเหมือนไม้เท้าทั่วไป

3. ใช้ประสาทสัมผัสจากการสั่นหรือเสียงจะมีสัญญาณเสียงและสั่นเมื่อเจอสิ่งกีดขวาง

ภาคผนวก ง.

ประวัติผู้จัดทำ

1. นรจ.วันชัย นาคนาคา เหล่าอิเล็กทรอนิกส์ (หัวหน้ากลุ่ม)
   ที่อยู่ 66 ม.7 ต.โคกกรวด อ.ปากพลี จ.นครนายก 26130
   จบจาก โรงเรียนมัธยมวัดป่ามะไฟ

2. นรจ.เฉลิมวงศ์ คำแพง เหล่าอิเล็กทรอนิกส์ (รองหัวหน้ากลุ่ม)
   ที่อยู่ 1ม.3 ต.ด่านม่วงคำ อ.โคกศรีสุพรรณ จ.สกลนคร 47280
   จบจาก วิทยาลัยเทคนิคสกลนคร

3. นรจ.ณัฐพงษ์ ศาสนาไทย เหล่าอิเล็กทรอนิกส์ (สมาชิก)
   ที่อยู่ 220/273 ม. 4 ต.ในคลองบางปลากด อ. พระสมุทรเจดีย์ จ. สมุทรปราการ 10290
   จบจาก โรงเรียน ป้องนาคราชสวาทยานนท์

4. นรจ.กฤตพจน์ สายสิม เหล่าอิเล็กทรอนิกส์ (สมาชิก)
   ที่อยู่ 117/3 ม.11 ต.โนนสัง อ.กันทรารมย์ 33130
   จบจาก โรงเรียนกันทรารมณ์

5. นรจ.พัฒนะ ทรัพย์สุนทร เหล่าอิเล็กทรอนิกส์ (สมาชิก)
   ที่อยู่ 227/81 หมู่.3 ซอย.ประสบโชค ต.สำโรง อ.พระประแดง จ.สมุทรปราการ 10130
   จบจาก โรงเรียนมัธยมวัดด่านสำโรง

6. นรจ.ณัฐกฤษฎิ์ จันทร์ประทัด เหล่าอิเล็กทรอนิกส์ (สมาชิก)
   ที่อยู่ 322 ถ.ทุ่งโพธิ์ ต.ในเมือง อ.เมือง จ.สุรินทร์ 32000
   จบจาก โรงเรียนเทศบาล3เทศบาลอนุสรณ์