เครื่องประกาศคำสั่งอัตโนมัติ

Automatic announcement machine

จัดทำโดย

นรจ.นราวิทย์	**ชูราษฎร์**
นรจ.สิทธิโชค	**แก้วสระแสน**
นรจ.ศุภพฎฐ์	**บุญซ้อน**
นรจ.วรภัทร	**สิงหะ**
นรจ.ปฏิวัติ	**นอกไธสง**

โครงงานนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรนักเรียนจ่า

พรรคพิเศษ เหล่าทหารช่างยุทธโยธา (อิเล็กทรอนิกส์)

โรงเรียนอิเล็กทรอนิกส์ กองวิทยาการ กรมอิเล็กทรอนิกส์ทหารเรือ ปีการศึกษา ๒๕๖๕

เครื่องประกาศคำสั่งอัตโนมัติ

Automatic announcement machine

จัดทำโดย

นรจ.นราวิทย์	**ชูราษฎร์**
นรจ.สิทธิโชค	**แก้วสระแสน**
นรจ.ศุภพฎฐ์	**บุญซ้อน**
นรจ.วรภัทร	**สิงหะ**
นรจ.ปฏิวัติ	**นอกไธสง**

ครูที่ปรึกษา

น.ต.ฐากร	**สร้อยมณี**
จ.อ.บุญญฤทธิ์	**บุญวัง**

คำนำ

รายงานงานเขียนเชิงวิชาการโครงงานสิ่งประดิษฐ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรนักเรียนจ่า พรรคพิเศษ เหล่าทหารช่างยุทธโยธา(อิเล็กทรอนิกส์) รร.อล.กวก.อล.ทร.จัดทำขึ้นเพื่อช่วยแก้ไขปัญหาเวลาในการพักประจำชั่วโมงของ นรจ.อล.กวก.อล.ทร. ซึ่งต้องใช้เวรยามที่รับผิดชอบประจำวันทำหน้าที่ประกาศพักประจำชั่วโมง ทำให้เวรยามประจำวันเรียนไม่ต่อเนื่องเพราะต้องขออนุญาตครูผู้สอนมาทำหน้าที่และช่วยแก้ไขปัญหาการล่าช้าในการประจำสถานีบนเรือ ที่อาจเกิดความผิดพลาดในการส่งคำสั่งล่าช้าของผู้มีหน้าที่ประกาศ

ในการนี้ทางคณะผู้จัดทำจึงได้พิจารณานำความรู้จากวิชาไมโครคอนโทรลเลอร์มาทำการสร้างเครื่องประกาศคำสั่งอัตโนมัติขึ้นมา เพื่อลดโอกาสที่จะเกิดข้อผิดพลาดจากการลืมมาเป่านกหวีดในแต่ละช่วงเวลา รวมถึงยังช่วยให้เวรยามที่ต้องทำหน้าที่ในวันนั้นๆ เรียนได้อย่างต่อเนื่อง และเพื่อเพิ่มความรวดเร็วในการประจำสถานีบนเรือ

คณะผู้จัดทำหวังว่าโครงงานฉบับนี้จะเป็นประโยชน์ต่อผู้ที่สนใจ หากมีข้อเสนอแนะเพื่อปรับปรุงแก้ไขกรุณาแจ้งผู้จัดทำ หรือแจ้งทาง รร.อล.กวก.อล.ทร. เพื่อดำเนินการปรับปรุงแก้ไขให้สมบูรณ์ยิ่งขึ้นต่อไป

คณะผู้จัดทำ
กลุ่มที่ ๑
นรจ.นราวิทย์ ชูราษฎร์
นรจ.สิทธิโชค แก้วสระแสน
นรจ.ศุภพฏฐ์ บุญซ้อน
นรจ.วรภัทร สิงหะ
นรจ.ปฏิวัติ นอกไธสง

ชื่อโครงงาน เครื่องประกาศคำสั่งอัติโนมัติ
ผู้จัดทำ นรจ.นราวิทย์ ชูราษฎร์
นรจ.สิทธิโชค แก้วสระแสน
นรจ.ศุภพฎฐ์ บุญซ้อน
นรจ.วรภัทร สิงหะ
นรจ.ปฏิวัติ นอกไธสง
ครูที่ปรึกษา น.ต.ฐากร สร้อยมณี
ปีการศึกษา 2565
จำนวนหน้า 62

บทคัดย่อ

การจัดทำโครงงานสิ่งประดิษฐ์ เครื่องประกาศคำสั่งอัตโนมัติ เป็นการศึกษาค้นคว้าเกี่ยวกับการนำความรู้จากวิชาไมโครคอนโทรลเลอร์มาประยุกต์ใช้งานในเป่านกหวีดพักประจำชั่วโมงของ นรจ.รร.อล.กวก.อล.ทร. และสามารถประยุกต์ใช้ในการประจำสถานีบนเรือ ซึ่งเมื่อถึงเวลาสามารถประกาศคำสั่งออกได้เองโดยไม่ต้องใช้เวรยามในการประกาศ และยังสามารถกดสวิตซ์เพื่อประกาศประจำสถานีภายในเรือได้ รวมถึงโครงงานนี้ยังช่วยลดการเสียเวลาเรียนของเวรยามประจำวัน และความล่าช้าจากการประกาศคำสั่งประจำสถานีบนเรือ

การศึกษาและการจัดสร้างโครงงานในครั้งนี้มีวัตถุประสงค์เพื่อไม่ให้เสียเวลาเรียนของเวรยามประจำวัน รวมถึงลดความล่าช้าในการประจำสถานีบนเรือ เพราะในการประกาศแต่ละครั้งต้องใช้เวรยามที่เข้าหน้าที่ในแต่ละวันทำการประกาศ จะทำให้เสียเวลาในการเรียนของเวรยามประจำวันรวมถึงความล่าช้าจากการประกาศคำสั่งสถานีบนเรือ

ผลจากการศึกษาและดำเนินการจัดสร้างโครงงานพบว่า โครงงานนี้สามารถทำให้เวรยามประจำวันเรียนต่อเนื่องประจำวิชานั้นๆและการประจำสถานีภายในเรือมีความรวดเร็วมากยิ่งขึ้น

น.ต.
(ฐากร สร้อยมณี)
ครูที่ปรึกษาโครงงาน

กิตติกรรมประกาศ

โครงงานสิ่งประดิษฐ์เรื่องเครื่องประกาศคำสั่งอัตโนมัติ นี้ได้รับการสนับสนุนเครื่องมือ และวัสดุฝึกในการทำโครงงานจากโรงเรียนอิเล็กทรอนิกส์ กองวิทยาการ กรมอิเล็กทรอนิกส์ทหารเรือ และได้รับแนวทางความรู้ในการดำเนินงานจากคณะครูที่ปรึกษาโครงงานกลุ่มที่ ๑ จนโครงงานสำเร็จลุล่วงไปได้ด้วยดี ทางคณะผู้จัดทำขอขอบพระคุณ น.อ.อนุสรณ์ วงศ์ปัญญา ผู้อำนวยการโรงเรียนอิเล็กทรอนิกส์ ที่สนับสนุนให้เกิดโครงงานสิ่งประดิษฐ์ของนักเรียนจ่า น.ต.สุชิน มุขศรี และผู้ที่เกี่ยวข้องทุกท่านที่ให้คำปรึกษาอันมีประโยชน์จนงานสำเร็จลุล่วงไปด้วยดี รวมทั้งครูที่ปรึกษาโครงงาน น.ต.ฐากร สร้อยมณี จ.อ.บุญญฤทธิ์ บุญวัง ที่คอยสนับสนุนการจัดหาเครื่องมือ และวัสดุอุปกรณ์ที่ใช้ทำโครงงาน และให้คำแนะนำให้คำปรึกษาเป็นประโยชน์ในการดำเนินการจัดทำโครงงานสิ่งประดิษฐ์นี้ให้ผ่านปัญหาข้อขัดข้องต่าง ๆ มาจนโครงงานเสร็จสมบูรณ์ และที่สำคัญทำให้นักเรียนซึ่งเป็นคณะผู้จัดทำได้มี ความรู้ความสามารถที่จะนำไปศึกษาต่อเพื่อพัฒนาตนเองในอนาคตต่อไป

คณะผู้จัดทำ
กลุ่มที่ ๑
นรจ.นราวิทย์ ชูราษฎร์
นรจ.สิทธิโชค แก้วสระแสน
นรจ.ศุภพฏฐ์ บุญซ้อน
นรจ.วรภัทร สิงหะ
นรจ.ปฏิวัติ นอกไธสง

สารบัญ

เรื่อง	**หน้า**
คำนำ	ก
บทคัดย่อ	ข
กิตติกรรมประกาศ	ค
บทที่ 1 บทนำ	
1.1 ความเป็นมาและความสำคัญของโครงงาน	1
1.2 วัตถุประสงค์ของโครงงาน	1
1.3 ขอบเขตของโครงงาน	1
1.4 ระยะเวลา	2
1.5 ผลที่คาดว่าจะได้รับ	2
บทที่ 2 ทฤษฎี และงานวิจัยที่เกี่ยวข้อง	
2.1 ทฤษฎีเสียง	3
2.2 Arduino Uno R3	6
2.3 Relay Module 5V 2 Channel	8
2.4 Arduino IDE	10
2.5 Real Time Clock DS3231 (RTC)	11
2.6 Maxxtronic 115	12
2.7 Push Button Switch	13
2.8 LCD 20*4 I2C	15
2.9 Light Emitting Diode (LED)	16
2.10 ทฤษฎีอื่นๆ ที่เกี่ยวข้อง	16
บทที่ 3 วิธีดำเนินโครงงาน	
3.1 แผนการดำเนินงาน	22
3.2 เครื่องมือ วัสดุอุปกรณ์ ในการทำโครงงาน	23
3.3 ขั้นตอนการทำโครงงาน	23
3.4 หลักการทำงานของเครื่องประกาศคำสั่งอัตโนมัติ	27
3.5 ผังการทำงานของเครื่องประกาศคำสั่งอัตโนมัติ	29
3.6 วงจรการทำงานของเครื่องประกาศคำสั่งอัตโนมัติ	31
บทที่ 4 ผลการทดลอง	
4.1 ผลการทดลอง	33

สารบัญ (ต่อ)

เรื่อง	หน้า
บทที่ 5 สรุปปัญหา และข้อเสนอแนะ	
5.1 สรุปผลการทำโครงงาน	36
5.2 ปัญหา	36
5.3 ข้อเสนอแนะ	36
ภาคผนวก	37-62
บรรณานุกรม	63
คณะผู้จัดทำ	64

สารบัญรูป

เรื่อง

รูปภาพที่ 2.1 แผนภูมิการได้ยิน
รูปภาพที่ 2.2 บอร์ด Arduino Uno R3 Pinout
รูปภาพที่ 2.3 Relay Module 5V 2 Channel
รูปภาพที่ 2.4 โปรแกรม Arduino IDE
รูปภาพที่ 2.5 Real Time Clock DS3231 (RTC)
รูปภาพที่ 2.6 Maxxtronic 115
รูปภาพที่ 2.7 Push Button Switch
รูปภาพที่ 2.8 โครงสร้างของ Push Button Switch
รูปภาพที่ 2.9 LCD 20*4 I2C
รูปภาพที่ 2.10 Light Emitting Diode (LED)
รูปภาพที่ 2.11 การขยายสัญญาณ
รูปภาพที่ 2.12 แอมป์หลอดสุญญากาศ
รูปภาพที่ 2.13 แอมป์ทรานซิสเตอร์
รูปภาพที่ 2.14 แอมป์มอสเฟส
รูปภาพที่ 2.15 แอมป์ที่ใช้วงจรรวม
รูปภาพที่ 3.1 ตัดแผ่นอะคริลิก
รูปภาพที่ 3.2 ประกอบโครงสร้าง
รูปภาพที่ 3.3 ติดตั้งอุปกรณ์ในโครงสร้าง
รูปภาพที่ 3.4 ทดสองชิ้นงาน
รูปภาพที่ 3.5 การใช้ Multimeter Digital
รูปภาพที่ 3.6 บล็อกไดอะแกรมการทำงาน
รูปภาพที่ 3.7 ผังการทำงานแบบ Automatic
รูปภาพที่ 3.8 ผังการทำงานแบบ Manual
รูปภาพที่ 3.9 วงจรการทำงานแบบ Automatic
รูปภาพที่ 3.10 วงจรการทำงานแบบ Manual

สารบัญตาราง

เรื่อง

ตารางที่ 2.1 ข้อมูลจำเพาะ Arduino Uno R3

ตารางที่ 3.1 แผนการดำเนินงาน

ตารางที่ 4.1 ผลการทดลองแบบ Automatic

ตารางที่ 4.2 ผลการทดลองแบบ Manual

ตารางที่ 4.3 ตารางเวลาที่สั่ง เปิด-ปิด พาวเวอร์แอมป์

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของโครงงาน

ในปัจจุบัน รร.อล.กวก.อล.ทร. ได้มีการพักประจำชั่วโมงของ นรจ.รร.อล.กวก.อล.ทร ซึ่งต้องใช้เวรยามที่รับผิดชอบประจำวันทำหน้าที่ประกาศพักประจำชั่วโมง ซึ่งต้องใช้เวรยามในแต่ละวันมาประกาศพักประจำชั่วโมง ทำให้เวรยามประจำวันเรียนไม่ต่อเนื่องเพราะต้องขออนุญาตครูผู้สอนมาทำหน้าที่ พร้อมทั้งปัจจุบันบนเรือได้มีการประจำสถานีต่างๆ ซึ่งต้องใช้คนที่มีหน้าที่ประกาศเป็นคนต่อไมค์ประกาศ จึงอาจเกิดความล่าช้าในการประจำสถานีต่างๆ

จากปัญหาข้างต้นดังกล่าว สามารถขจัดให้หมดไปได้โดยการสร้างเครื่องประกาศคำสั่งอัตโนมัติ เข้ามาช่วยในการประกาศแทนการใช้เวรยามในวันนั้นๆ ทางกลุ่มโครงงานจึงได้คิดจะประดิษฐ์เครื่องประกาศคำสั่งอัตโนมัติ ที่สามารถประกาศคำสั่งต่างๆตามเวลาได้ด้วยตัวเอง อีกทั้งยังมีปุ่มกดสวิตช์ให้คนที่ทำหน้าที่ประกาศคำสั่งในเรือได้กดเพื่อเพิ่มความรวดเร็วในการประจำสถานี

1.2 วัตถุประสงค์ของโครงงาน

1.2.1 เพื่อสร้างเครื่องประกาศคำสั่งอัตโนมัติ

1.2.2 เพื่อไม่ให้เสียเวลาเรียนของเวรยามประจำวันที่มีหน้าที่เป้านกหวีดประกาศ

1.2.3 เพื่อเพิ่มความสะดวกรวดเร็วในการประกาศประจำสถานีภายในเรือ

1.2.4 เพื่อลดปัญหาการ ปิด-เปิด ของเครื่อง Amplifier บ่อยเกินไป

1.3 ขอบเขตของโครงงาน

1.3.1 สร้างเครื่องประกาศคำสั่งอัตโนมัติ

1.3.2 ใช้สวิตช์แทนการใช้ไมโครโฟน

1.3.3 เครื่องประกาศอัตโนมัติให้เสียงออกมาตามที่กำหนด

1.3.4 เครื่องประกาศคำสั่งอัตโนมัติอ้างอิงมาจากเวลาปัจจุบัน

1.3.5 สร้างการ ปิด-เปิด Amplifier ให้ทำงานอัตโนมัติ เมื่อมีคำสั่งให้ใช้งาน

1.4 ระยะเวลา

ระยะเวลาในการทำโครงงานตั้งแต่วันที่ 30 ม.ค. 2566 – 24 มี.ค.2566 (รวม 8 สัปดาห์)

1.5 ผลที่คาดว่าจะได้รับ

1.5.1 ได้เครื่องประกาศคำสั่งอัตโนมัติ

1.5.2 สามารถนำเครื่องประกาศคำสั่งอัตโนมัติไปใช้ใน รร.อล.กวก.อล.ทร.

1.5.3 ได้เครื่องอำนวยความสะดวกมาใช้ภายในเรือ

1.5.4 สามารถลดปัญหาการ ปิด-เปิด Amplifier บ่อยครั้งจนเกินไปลดการเกิดความเสียหายได้

บทที่ 2
เอกสารและงานวิจัยที่เกี่ยวข้อง

เครื่องประกาศคำสั่งอัตโนมัติ มีเอกสารและงานวิจัยที่เกี่ยวข้อง ดังนี้

2.1 ทฤษฎีเสียง

2.2 Arduino Uno R3

2.3 Relay Module 5V 2 Channel

2.4 Real Time Clock DS3231 (RTC)

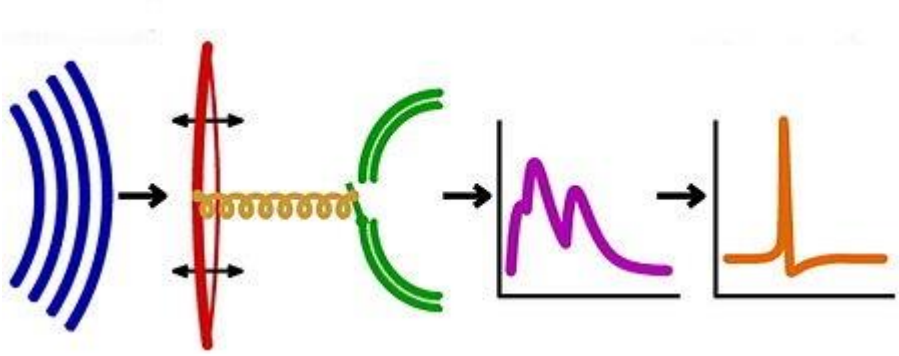
2.5 บอร์ด Maxxtronic 115

2.6 Push Button Switch

2.7 LCD 20*4 I2C

2.8 ทฤษฎีอื่นๆ ที่เกี่ยวข้อง

2.1 ทฤษฎีเสียง



รูปภาพที่ 2.1 แผนภูมิแสดงการได้ยิน
ที่มา https://th.wikipedia.org/wiki/

2.1.1 ความหมายของเสียง

เป็นคลื่นเชิงกลที่เกิดจากการสั่นสะเทือนของวัตถุ เมื่อวัตถุสั่นสะเทือน ก็จะทำให้เกิดการอัดตัวและขยายตัวของคลื่นเสียง และถูกส่งผ่านตัวกลาง เช่น อากาศ ไปยังหู แต่เสียงสามารถเดินทางผ่านสสารในสถานะก๊าซ ของเหลว และของแข็งก็ได้ แต่ไม่สามารถเดินทางผ่านสุญญากาศได้

เมื่อการสั่นสะเทือนนั้นมาถึงหู มันจะถูกแปลงเป็นพัลส์ประสาท ซึ่งจะถูกส่งไปยังสมอง ทำให้เรารับรู้และจำแนกเสียงต่างๆ ได้

2.1.2 การเกิดเสียง

เริ่มเกิดขึ้นเมื่อวัตถุหรือแหล่งกำเนิดเสียงมีการสั่นสะเทือนส่งผลต่อการเคลื่อนที่ของโมเลกุลของอากาศที่อยู่โดยรอบ กล่าวคือโมเลกกุลของอากาศเหล่านี้จะเคลื่อนที่จากตำแหน่งเดิมไปชนกับโมเลกุลที่อยู่ถัดไป ก่อให้เกิดการถ่ายโอนโมเมนตัมจากโมเลกุลที่มีการเคลื่อนที่ให้กับโมเลกุลที่อยู่ในส ภาวะปกติ จากนั้นโมเลกุลที่ชนกันนี้จะแยกออกจากกันโดยโมเลกุลที่เคลื่อนที่มาจะถูกดึงกลับไปยังตำแหน่งเดิมด้วยแรงปฏิกิริยาและโมเลกุลที่ได้รับการถ่ายโอนพลังงานจะเคลื่อนที่ไปชนกับโมเลกุลที่อยู่ถัดไป ปรากฏการณ์นี้จะเกิดขึ้นสลับกันไปมาได้เมื่อสื่อกลาง (ในที่นี้คืออากาศ) มีคุณสมบัติของความยืดหยุ่น การเคลื่อนที่ของโมเลกุลอากาศนี้จึงเกิดเป็นคลื่นเสียง

2.1.3 คุณลักษณะของเสียง

คุณลักษณะเฉพาะของเสียง ได้แก่ ความยาวช่วงคลื่น แอมพลิจูด ความเร็ว และ ความเข้มเสียง เสียงแต่ละเสียงมีความแตกต่างกัน เสียงสูง-เสียงต่ำ, เสียงดัง-เสียงเบา, หรือคุณภาพของเสียงลักษณะต่าง ๆ ทั้งนี้ขึ้นอยู่กับแหล่งกำเนิดเสียง และจำนวนรอบต่อวินาทีของการสั่นสะเทือน คุณลักษณะของเสียงประกอบด้วย 5 ลักษณะได้แก่

1) ความถี่

ระดับเสียง (pitch) หมายถึง เสียงสูงเสียงต่ำ สิ่งที่ทำให้เสียงแต่ละเสียงสูงต่ำแตกต่างกันนั้น ขึ้นอยู่กับความเร็วในการสั่นสะเทือนของวัตถุ วัตถุที่สั่นเร็วเสียงจะสูงกว่าวัตถุที่สั่นช้า โดยจะมีหน่วยวัดความถี่ของการสั่นสะเทือนต่อวินาที เช่น 60 รอบต่อวินาที, 2,000 รอบต่อวินาที เป็นต้น และนอกจาก วัตถุที่มีความถี่ในการสั่นสะเทือนมากกว่า จะมีเสียงที่สูงกว่าแล้ว หากความถี่มากขึ้นเท่าตัว ก็จะมีระดับเสียงสูงขึ้นเท่ากับ 1 ออกเตฟ (octave) ภาษาไทยเรียกว่า 1 ช่วงคู่แปด

2) ความยาวช่วงคลื่น

ความยาวช่วงคลื่น (wavelength) หมายถึง ระยะทางระหว่างยอดคลื่นสองยอดที่ติดกันซึ่งเกิดขึ้นระหว่างการอัดตัวของคลื่นเสียง (คล้ายคลึงกับยอดคลื่นในทะเล) ยิ่งความยาวช่วงคลื่นมีมาก ความถี่ของเสียง (ระดับเสียง) ยิ่งต่ำลง

3) ความกว้างช่วงคลื่น

ความกว้างช่วงคลื่น (bandwidth) หมายถึง ขนาดของวงคลื่นสองวงที่แผ่กันซึ่งเกิดขึ้นเมื่อความยาวของคลื่นเสียงมีน้อย ยิ่งความกว้างช่วงคลื่นมีมาก ความถี่ของเสียง (ระดับเสียง) ยิ่งสูงขึ้น

4) ความดันเสียง

หมายถึง ค่าความดันของคลื่นเสียงที่เปลี่ยนแปลงไปจากความดันบรรยากาศปกติ ซึ่งค่าความดันที่เปลี่ยนแปลงมากที่สุด คือ ค่าความสูงคลื่นหรือแอมพลิจูด การตอบสนองของหูต่อความดันเสียงไม่ได้มีลักษณะเป็นเส้นตรง แต่มีความสัมพันธ์นลักษณะของลอกาลิทึม (Logarithm) ดังนั้น ค่าระดับความดันเสียง ที่อ่านได้จากการตรวจวัดโดยเครื่องวัดเสียงนั้น เป็นค่าทีได้จากการเปรียบเทียบกับความดันเสียงอ้างอิงแล้ว มีหน่วยวัดเป็น เดซิเบล (decibel : dB)

5) แอมพลิจูด

แอมพลิจูด (amplitude) หมายถึง ความสูงระหว่างยอดคลื่นและท้องคลื่นของคลื่นเสียง ที่แสดงถึงความเข้มของเสียง (Intensity) หรือความดังของเสียง (Loudness) ยิ่งแอมพลิจูดมีค่ามาก ความเข้มหรือความดังของเสียงก็ยิ่งเพิ่มขึ้น

2.1.4 ประเภทของเสียง

แบ่งตามลักษณะการเกิดได้ 3 ลักษณะ

1) เสียงดังแบบต่อเนื่อง (continuous Noise) เป็นเสียงดังที่เกิดขึ้นอย่างต่อเนื่อง จำแนกออกเป็น 2 ลักษณะ คือ เสียงดังต่อเนื่องแบบคงที่ (steady-state Noise) และเสียงดีงต่อเนื่องที่ไม่คงที่ (Non steady state Noise)

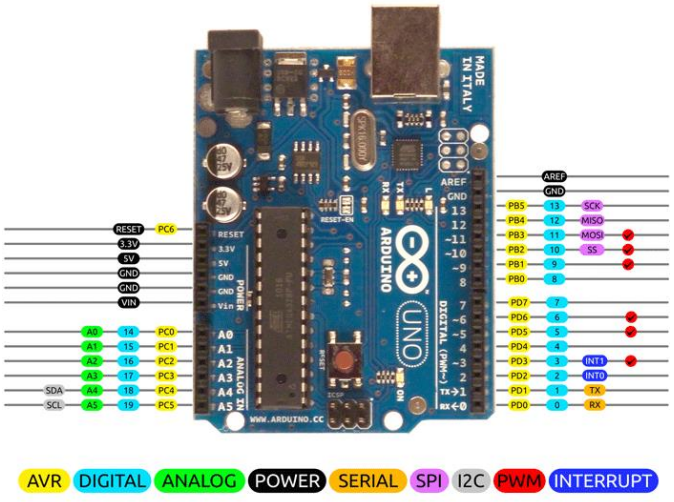
   - เสียงดังต่อเนื่องแบบคงที่ (Steady-state Noise) เป็นลักษณะเสียงดังต่อเนื่องที่มีระดับเสียง เปลี่ยนแปลง ไม่เกิน 3 เดซิเบล เช่น เสียงจากเครื่องทอผ้า เครื่องปั่นด้าย เสียงพัดลม เป็นต้น
   - เสียงดังต่อเนื่องที่ไม่คงที่ (Non-steady state Noise) เป็นลักษณะเสียงดังต่อเนื่องที่มี ระดับเลียงเปลี่ยนแปลงเกินกว่า 10 เดซิเบล เช่น เสียงจากเลื่อยวงเดือน เครื่องเจียร เป็นต้น

2) เสียงดังเป็นช่วง ๆ (Intermittent Noise) เป็นเสียงที่ดังไม่ต่อเนื่อง มีความเงียบหรือเบากว่าเป็นระยะ ๆ ลลับไปมา เช่น เสียงเครื่องปั๊ม/อัดลม เสียงจราจร เสียงเครื่องบินที่บินผ่านไปมา เป็นต้น

3) เสียงดังกระทบ หรือ กระแทก (Impact or Impulse Noise) เป็นเสียงที่เกิดขึ้นและสิ้นสุดอย่างรวดเร็ว ในเวลาน้อยกว่า 1 วินาที มีการเปลี่ยนแปลงของเสียงมากกว่า 40 เดซิเบล เช่น เสียงการตอกเสาเข็ม การปั๊มชิ้นงาน การทุบเคาะอย่างแรง เป็นต้น

2.2 Arduino Uno R3

Arduino Uno R3 คือ บอร์ด UNO เป็นบอร์ดไมโครคอนโทรลเลอร์ที่ใช้ไอซีเบอร์ ATmega328P เป็นไอซีหลัก ซึ่งภายในตัวไอซีจะมีขา INPUT และ OUTPUT แบบดิจิตอลจำนวน 14 ขา (สามารถใช้เป็นขา PWM output จำนวน 6 ขา) นอกจากนั้นยังมีขาแบบ Analog ไว้ให้ใช้งานจำนวน 6 ขา บนบอร์ดยังมีสิ่งอำนวยความสะดวกมากมายสำหรับผู้เริ่มต้น เช่น ขั้ว USB แบบ B ใช้ในการติดต่อสื่อสารกับคอมพิวเตอร์, ขั้ว DC-IN, ขั้วต่อ ICSP และปุ่ม Reset เป็นต้น ตัวบอร์ดสามารถเลือกใช้แหล่งจ่ายไฟได้จากขั้ว USB หรือขั้ว DC-IN ก็ได้ ทำให้สะดวกมากยิ่งขึ้นในการใช้งาน บอร์ด UNO R3 สามารถใช้ร่วมกับโปรแกรม Arduino IDE ได้ทุกเวอร์ชั่น ขนาดของ Flash Memory คือ 32 กิโลไบท์ (โดยถูกจองด้วยโปรแกรม bootloader เป็นจำนวน 0.5 กิโลไบท์) ขนาดของ SRAM คือ 2 กิโลไบท์ และขนาดของ EEPROM คือ 1 กิโลไบท์



รูปภาพที่ 2.2 บอร์ด Arduino Uno R3 Pinout

ที่มา https://components101.com/microcontrollers/arduino-uno

2.2.1 ข้อมูลจำเพาะ

ตารางที่ 2.1 ข้อมูลจำเพาะ Arduino Uno R3

ชิปไอซีไมโครคอนโทรเลอร์	ATmega328P
ใช้แรงดันไฟฟ้า	5V
รองรับการจ่ายแรงดันไฟฟ้า (ที่แนะนำ)	7 – 12V
รองรับการจ่ายแรงดันไฟฟ้า (ที่จำกัด)	6 – 20V
พอร์ต Digital I/O	14 พอร์ต (มี 6 พอร์ต PWM output)
พอร์ต Analog Input	6 พอร์ต
กระแสไฟที่จ่ายได้ในแต่ละพอร์ต	40mA
กระแสไฟที่จ่ายได้ในพอร์ต 3.3V	50mA
พื้นที่โปรแกรมภายใน	32KB พื้นที่โปรแกรม, 500B ใช้โดย Booloader
พื้นที่แรม	2KB
พื้นที่หน่วยความจำถาวร (EEPROM)	1KB
ความถี่คริสตัล	16MHz
ขนาด	68.6x53.4 mm
น้ำหนัก	25 กรัม

2.2.2 ขาต่อใช้งาน Arduino Uno R3

1) ขา INPUT/OUTPUT แบบ DIGITAL จะมีจำนวนขาใช้งานทั้งหมด 14 ขา ซึ่งสามารถกำหนดให้เป็นขา INPUT หรือขา OUTPUT ก็ได้ ตามการเขียนโปรแกรม โดยขานี้จะทำงานที่ระดับแรงดัน 5 โวลท์ดีซี กระแสที่สามารถจ่ายและรับได้ 20 มิลลิแอมป์ โครงสร้างภายในขาจะมีตัวต้านทานแบบพูลอัพต่ออยู่ ค่าประมาณ 20-50 กิโลโอห์ม นอกจากนั้นในบางขายังสามารถกำหนดฟังก์ชั่นพิเศษได้ เช่น

- ขา 0 (RX) และขา 1 (TX) ใช้ในการติดต่อสื่อสารแบบอนุกรมในระดับ TTL
- ขา 2 และขา 3 เป็นขา Interrupt จากภายนอก ตามฟังก์ชั่น attachInterrupt()
- ขา PWM ได้แก่ ขา 3,5,6,9,10 และ 11 เป็นขา OUTPUT ขนาด 8 บิต ตามฟังก์ชั่น analogWrite()
- ขา 10 (SS), ขา 11 (MOSI), ขา 12 (MISO) และขา 13 (SCK) เป็นขาที่ใช้ในการติดต่อสื่อสารแบบ SPI โดยการใช้ไลบอรี่ SPI

- ขา 13 เป็นขาที่ต่ออยู่กับ LED บนบอร์ด โดยถ้ากำหนดเป็นขา OUTPUT และมีค่าเป็น High ตัว LED จะติด แต่ถ้ากำหนดเป็น Low ตัว LED จะดับ

2) ขา INPUT แบบ ANALOG จะมีจำนวนขาใช้งานทั้งหมด 6 ขา ซึ่งในแต่ละขาจะมีระดับแรงดันสูงสุด 5 โวลท์ เมื่อเทียบกับกราวน์ และมีความละเอียดขนาด 10 บิต

3) ขา Vin เป็นขารับแหล่งจ่ายไฟจากภายนอก ถ้าทำการจ่ายไฟเข้าที่ขานี้ ไม่ควรทำการจ่ายไฟผ่านขั้ว USB และขั้ว DC-IN

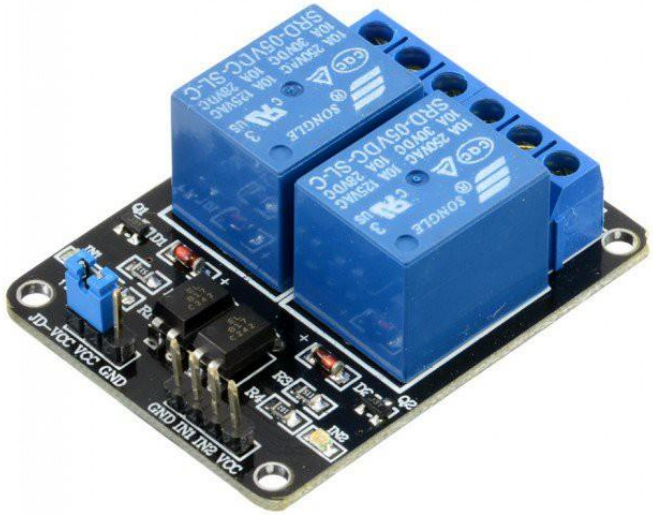
4) ขา Vout 5V เป็นขาจ่ายไฟบวกขนาด 5 โวลท์ดีซี

5) ขา Vout 3.3V เป็นขาจ่ายไฟบวกขนาด 3.3 โวลท์ดีซี กระแสสูงสุด 50 มิลลิแอมป์

6) ขา GND เป็นขากราวน์ของวงจร

2.3 Relay Module 5V 2 Channel

โมดูลรีเลย์ 2 ช่อง 5V (2 Channel Relay Module) เป็นโมดูลที่ใช้ควบคุมโหลดได้ทั้งแรงดันไฟฟ้า DC และ AC ซึ่งโหลดสูงสุด (Maximum Load) คือ AC 250V/10A, DC 30V/10A โดยใช้สัญญาณในการควบคุมการทำงานด้วยสัญญาณโลจิก TTL ทำงานด้วยสัญญาณแบบ Active Low, กระแสขับรีเลย์ (Drive Current) 15-20mA., มีการออกแบบให้เป็น Isolate ด้วย Optocoupler, มี LED แสดงสถานะ Relay สามารถนำไปประยุกต์ใช้งาน PLC Control, บ้านอัจฉริยะ, ใช้ในโรงงานอุตสาหกรรม หรืองานอื่นๆ ขึ้นอยู่กับการเขียนโปรแกรมและการต่อใช้งานภายนอก สามารถเชื่อมต่อใช้งานกับบอร์ด Rasberry Pi, Arduino, ARM, MCS-51, AVR, PIC, 8051, DSP, MSP430, TTL logic



รูปภาพที่ 2.3 Relay Module 5V 2 Channel
ที่มา https://th.cytron.io/p-2-channel-dc-5v-relay-module

2.3.1 ข้อมูลจำเพาะ

- ไฟเลี้ยงโมดูลรีเลย์ VCC = 5VDC.
- ควบคุมโหลดได้ทั้งแรงดันไฟฟ้า AC ได้สูงสุด 250VAC 10A หรือ แรงดันไฟฟ้า DC ได้สูงสุด 30VDC 10A (Maximum Load)
- ระดับสัญญาณอินพุทควบคุมแบบ TTL ทำงานด้วยสัญญาณแบบ Active Low
- กระแสขับรีเลย์ (Drive Current) 15-20mA.
- มีการออกแบบให้เป็น Isolate ด้วย Optocoupler

- มี LED แสดงสถานะ Relay
- โมดูลขนาด 3.85cm.(กว้าง) x 5.05cm.(ยาว) x 1.85cm.(สูง)

2.4 Arduino IDE

Arduino IDE คือ ซอฟต์แวร์เครื่องมือสำหรับพัฒนาโปรแกรมด้วยภาษา C / C++ สำหรับควบคุมบอร์ด Arduino และเครื่องมือต่าง ๆ ที่จำเป็นสำหรับใช้งานบอร์ด Arduino เช่น Serial Monitor , Compile ,Libraries ฯลฯ

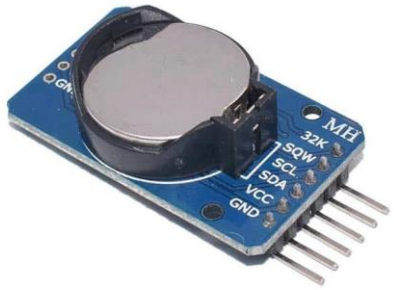
IDE (Integrated Development Environment) คือ ซอฟต์แวร์เครื่องมือสำหรับพัฒนาโปรแกรม ซึ่งมีสิ่งอำนวยความสะดวกต่าง ๆ เช่น RUN ,Compile ,DEBUG ซึ่งมี GUI ที่ถูกออกแบบมาให้มีสภาวะแวดล้อม (Environment) เหมาะสมการพัฒนาโปรแกรม โดยหน้าที่หลักของ IDE คือการเขียนไฟล์ เปิดไฟล์ บันทึกไฟล์ ทดสอบการทำงาน จัดเตรียมข้อมูล รวมถึงจัดการ Directory สำหรับภาษานั้น ๆ ที่รองรับ

รูปภาพที่ 2.4 โปรแกรม Arduino IDE

ที่มา https://www.techtonions.com/getting-familiar-with-arduino-ide/

2.5 Real Time Clock DS3231 (RTC)

DS3231 module เป็นโมดูนาฬิกาแบบเวลาจริง RTC (Real Time Clock) ที่มีความถูกต้องแม่นยำสูง เพราะข้างในมีวงจรวัดอุณหภูมิ เพื่อนำอุณหภูมิจากสภาพแวดล้อมมาคำนวณชดเชยความถี่ของ Crystal ที่ถูกรบกวนจากอุณหภูมิภายนอก มาพร้อมแบตเตอร์รี่ ใช้งานได้แม้ไม่มีแหล่งจ่ายไฟจากภายนอก สามารถตั้งค่า วัน เวลา ได้อย่างง่าย มีไลบารีมาพร้อมใช้งาน สามารถเลือกแสดงผลเวลาแบบ 24 ชั่วโมงหรือแบบ 12 ชั่วโมงก็ได้ นอกจากจะแสดงวันและเวลาได้อย่างแม่นยำแล้ว โมดูลนี้ยังสามารถแสดงอุณหภูมิภายนอกได้ เป็นเหมือนนาฬิกาดิจิตอลที่บอกอุณหภูมิได้ด้วย



รูปภาพที่ 2.5 Real Time Clock DS3231 (RTC)
ที่มา https://www.cybertice.com/article/40/

2.6 Maxxtronic 115

วงจรนี้ เป็นวงจรเล่นไฟล์เสียงประเภท MP3 ที่สามารถต่อเซ็นเซอร์ได้หลายรูปแบบ เพื่อเป็นตัวควบคุมการทำงาน วงจรนี้สามารถนำไปติดตั้งได้หลากหลายงาน เช่น เป็นวงจรเตือนภัย, วงจรต้อนรับ เป็นต้น

รูปภาพที่ 2.6 Maxxtronic 115
ที่มา https://www.futurekitshop.com/product/553/maxxtronic-mxa115-

2.6.1 ข้อมูลจำเพาะ

- ใช้ไฟเลี้ยงวงจร 5 โวลท์ดีซี

- ขณะสแตนด์บาย กินกระแสสูงสุด 25 มิลลิแอมป์

- ขณะทำงาน กินกระแสสูงสุด 300 มิลลิแอมป์ ที่ลำโพง 8 โอห์ม 0.25 วัตต์

- สามารถเลือกการเล่นข้อความได้ 2 แบบ คือ แบบเล่นเฉพาะข้อความที่ 1 เท่านั้นและแบบเล่นทีละข้อความต่อการทริก 1 ครั้ง (ไล่ลำดับ)

- สามารถต่อเซ็นเซอร์ภายนอกได้ เพื่อเริ่มการทำงานของวงจร

- ตัววงจรรองรับไฟล์ MP3 และ WMV รวมทั้งเป็นไฟล์ระบบ FAT16 และ FAT32

- สามารถใช้ได้กับการ์ด MicroSD หน่วยความจำสูงสุดที่รองรับ 32GB

- มีวงจรขยายเสียงขนาด 3 วัตต์ อยู่ภายในบอร์ด

- มีจุดต่อสำหรับต่อวงจรขยายเสียงภายนอกได้

- ขนาดแผ่นวงจรพิมพ์ : 2.45x3.01 นิ้ว (แผ่นวงจรควบคุม)

2.7 Push Button Switch

Push Button Switch หรือที่เรียกกันว่าสวิตซ์ปุ่มกด เป็นอุปกรณ์ทางไฟฟ้า ซึ่งทำหน้าที่ตัดและต่อวงจรทางไฟฟ้าและ ใช้ในการควบคุมการทำงานของมอเตอร์ หรือการทำงานของเครื่องจักรต่างๆ เป็นเหมือนอุปกรณ์พื้นฐาน ใช้ได้กับอุตสาหกรรมทั่วไป มีทั้งแบบมีไฟ และทึบแสง

รูปภาพที่ 2.7 Push Button Switch
ที่มา https://mall.factomart.com/principle-of-push-button-switch/

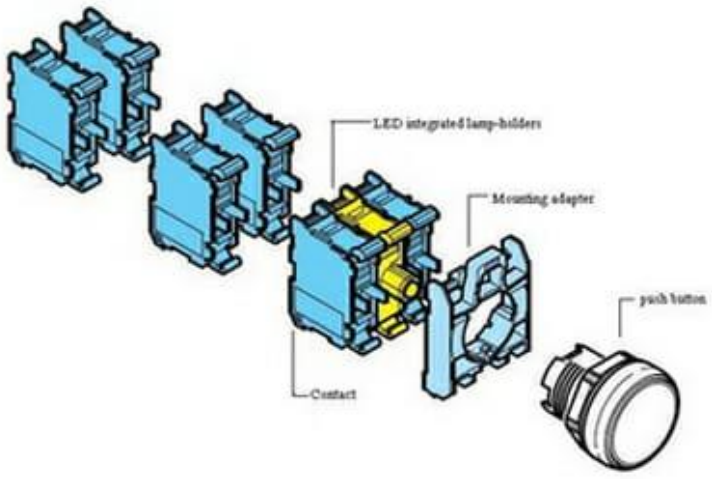
2.7.1 โครงสร้างของ Push Button Switch

- ปุ่มกดทำด้วยโลหะหรือพลาสติกซึ่งจะมีหลายหลายสีให้เลือกใช้งาน

- ฐานยึดระหว่างปุ่มกดและตัวล็อกหน้าสัมผัส โดยจะมีเกลียวที่ฐานเพื่อไว้สำหรับยึดอุปกรณ์กับชิ้นงานด้วย

- หน้าสัมผัส NO และ NC

- หลอดไฟ LED ที่ใช้แสดงสถานะ



รูปภาพที่ 2.8 โครงสร้างของ Push Button Switch
ที่มา https://mall.factomart.com/what-is-a-push-button-switch/

2.7.2 หลักการทำงานของ Push Button Switch

Push button Switch มีทั้งหมด 2 ชนิด ดังนั้น หลักการทำงานจึงมีความแตกกต่างกัน ดังนี้

- แบบกดติด ปล่อยดับ เมื่อมีการกด Push button Switch หน้าสัมผัสดังกล่าวจะเปลี่ยนสถานะ จาก NO เป็น NC หรือ จาก NC จะเป็น NO แต่เมื่อปล่อยมือออกจาก Push button Switch หน้าสัมผัสจะกลับสู้สภาวะปกติในแหน่งเดิมโดยมีแรงผลักดันจากสปริงให้ Push button Switchเข้าสู้สภาวะปกติ
- แบบกดติด กดดับ เมื่อมีการกด Push button Switch หน้าสัมผัสดังกล่าวจะเปลี่ยนสถานะ จาก NO เป็น NC หรือจาก NC จะเป็น NO แต่เมื่อปล่อยมือออกจาก Push button Switch หน้าสัมผัสจะถูกล็อกไว้โดยกลไกลของสวิตซ์ ซึ่งสามารถกลับสู้สภาวะปกติในแหน่งเดิมได้โดยโดยกด Push button Switchอีกครั้งทำให้คลายล็อก จะมีแรงผลักดันจากสปริงให้Push button Switchเข้าสู้สภาวะปกติ

2.8 LCD 20*4 I2C

คำว่า LCD ย่อมาจากคำว่า Liquid Crystal Display ซึ่งเป็นจอที่ทำมาจากผลึกคริสตัลเหลว หลักการคือ ด้านหลังจอจะมีไฟส่องสว่าง หรือที่เรียกว่า Backlight อยู่ เมื่อมีการปล่อยกระแสไฟฟ้าเข้าไปกระตุ้นที่ผลึก ก็จะทำให้ ผลึกโปร่งแสง ทำให้แสงที่มาจากไฟ Backlight แสดงขึ้นมาบนหน้าจอ ส่วนอื่นที่โดนผลึกปิดกั้นไว้จะไม่สว่าง ผลึกมีสีที่ แตกต่างกันตามสีของผลึกคริสตัล เช่นสีเขียว หรือ สีฟ้า ทำให้เมื่อมองไปที่จอก็จะพบกับตัวหนังสือแล้วพบกับ พื้นหลังสีต่างๆกัน

รูปภาพที่ 2.9 LCD 20*4 I2C
ที่มา https://www.spmicrotech.com/product/lcd-character-display-2004-yellow-with-i2c/

2.8.1 ข้อมูลจำเพาะ

- LCD แบบตัวอักษรขนาด 20 ตัวอักษร 4 บรรทัด พื้นสีน้ำเงิน ตัวอักษรสีขาว (STN Negative Blue)
- การำชื่อมต่อแบบ I2C-bus ท้าให้สามารถต่อร่วมกันได้ 8 ตัวบนบัสเดียวกัน (เลือก Address ต่างกัน)
- ใช้ไอซีขยายพอร์ตำบอร์ PCF8574A หรือ PCF8574 ในการำชื่อมต่อกับจอ LCD
- ใช้แรงดันไฟเลี้ยง 5 VDC

2.8.2 ส่วนประกอบ

- ขาต่อใช้งานมี 4 ขา มีดังนี้
    1) ขา GND ขากราวด์
    2) ขา VCC ขาไฟเลี้ยงใช้แรงดัน 5 VDC
    3) ขา SDA ขาสัญญาณ DATA ของระบบ I2C-bus

4) ขา SCL ขาสัญญาณ CLOCK ของระบบ I2C-bus

- ตัวต้านทานปรับค่าได้ สำหรับปรับความเข้มขอเจอ LCD
- จั้มเปอร์เลือก Address (A0-A2) ขอเจอ LCD โดยจั้มเปอร์นี้ถ้าไม่เชื่อมต่อจะได้ค่าลอจิก 1 และถ้าเชื่อมต่อจะ ได้ค่าลอจิก 0 ซึ่งปกติจะไม่ได้บัดกรีไว้ทำาให้ Address เริ่มต้นคือ 0x3F (A2=1 , A1=1 , A0=1) แต่ถ้าผู้ใช้ ต้องการเปลี่ยน Address ก็สามารถเปลี่ยนได้ 8 ค่า คือ 0x38-0x3F
- ไอซีขยายพอร์ตำบอร์ PCF8574A หรือ PCF8574
- LED POWER แสดงการจ่ายไฟเลี้ยงขอเจอ LCD
- จั้มเปอร์สำหรับเปิดปิดการจ่ายไฟ ของไฟส่องสว่างด้านหลังจอ LCD ถ้าต่อไว้จะเป็นการเปิดไฟเลี้ยงไฟส่องสว่างด้านหลังจอ LCD ตัวอย่างการใช้งาน LCD 20 x 4

2.9 Light Emitting Diode (LED)

LED คือ ไดโอดเปล่งแสง ย่อมาจากคำว่า (Light-Emitting Diode) ซึ่งสามารถเปล่งแสงออกมาได้ แสงที่เปล่ง ออกมาประกอบด้วยคลื่นความถี่เดียวและเฟสต่อเนื่องกัน ซึ่งต่างกับแสงธรรมดาที่ตาคนมองเห็น โดยหลอด LED สามารถเปล่งแสงได้เมื่อจ่ายกระแสไฟฟ้าเข้าเพียงเล็กน้อยเท่านั้น

รูปภาพที่ 2.10 Light Emitting Diode (LED)

ที่มา https://shopee.com.my/5mm-Light-Emitting-Diode-%28LED%29-2-Pins-i.61559496.1630611078

2.10 ทฤษฎีอื่น ๆ ที่เกี่ยวข้อง

2.10.1 GPIO

GPIO ย่อมาจาก General Purpose Input/Output คือ พอร์ตเอนกประสงค์ซึ่งสามารถเป็นได้ทั้ง อินพุต และ เอาต์พุต เราสามารถควบคุม คอนโทรลให้เป็นลอจิก “1” หรือ ลอจิก “0” ได้ โดยควบคุมได้แต่ละ pin

ในการใช้งานขา GPIO แบบดิจิตอล จำเป็นต้องมีการควบคุมหรือบอกให้ขา GPIO นั้น ๆ ได้รู้ว่าต้องการจะให้ใช้รับค่าดิจิตอลเข้ามา หรือจะให้เขียนค่าออกไป ซึ่งจะใช้คำสั่ง pinMode() ในการกำหนด คำสั่ง pinMode() มีรูปแบบการใช้งาน ดังนี้

void pinMode(pin ,mode);

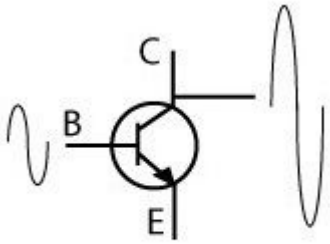
ฟังก์ชั่นมีค่าพารามิเตอร์ 2 ตัว คือ

1) pin – กำหนดหมายเลขขา GPIO ที่ต้องการควบคุม

2) mode – โหมดที่ต้องการกำหนดให้ขา GPIO ซึ่งสามารถเป็นได้ ดังนี้

- INPUT – กำหนดให้ขา GPIO มีสถานะเป็นอินพุต รอรับค่าเข้ามา
- OUTPUT – กำหนดให้ขา GPIO มีสถานะเป็นเอาต์พุต รอเขียนค่าออกไป

2.10.2 หลักการขยายของ Amplifier



รูปภาพที่ 2.11 การขยายสัญญาณ
ที่มา https://www.audio2home.com

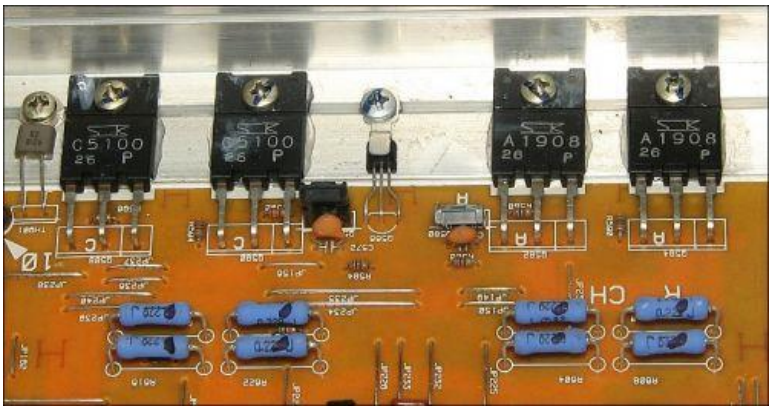
เครื่องขยายเสียง หรือเครื่องขยายสัญญาณเสียง โดยคำว่าเครื่องขยายสัญญาณเสียง แอมพลิฟายเออร์ (Amplifier) หรืออาจเรียกว่า เพาเวอร์แอมพลิฟายเออร์ (Power Amplifier) นั้น เป็นวงจรเครื่องหรืออุปกรณ์ทางระบบไฟฟ้า ที่ใช้ทำหน้าที่ ขยาย สัญญาณเสียง ขนาดเล็ก ให้มีขนาดสัญญาณเสียงที่สูงขึ้น หรือความดังมากขึ้น โดยตามอุดมคติ จะต้องขยายเสียงแล้วให้เสียงเหมือนจริง แต่มีความดังมากขึ้นนั่นเอง โดยเครื่องขยายเสียง,เครื่องขยายสัญญาณเสียง หรือ เพาเวอร์แอมพลิฟายเออร์ (Power Amplifier) นั้นมีหลักการขั้นพื้นฐานมาจากการใช้ ทรานซิสเตอร์ (Transistor)TR.หรือสารกึ่งตัวนำ ชนิดอื่นๆ หรือ หลอดสูญญากาศ (Vacum tube) มาเป็นตัวทำหน้าที่ขยายสัญญาณเสียงเล็กๆให้ ดังมากขึ้น โดยในที่นี้จะขออธิบายพื้นฐานการขยายโดยใช้ ทรานซิสเตอร์ (Transistor) TR. โดยการป้อนสัญญาณเข้าขั้วบี B ของ ทรานซิสเตอร์ (Transistor) และจะได้สัญญาณออกทางขั้วซี C เป็นสัญญาณที่มีขนาดสูงขึ้น และนี่ก็คือแนวคิดในการประยุกต์สำหรับการออกแบบสร้างเครื่องขยายเสียงหลากหลายรูปแบบ โดยเอาอุปกรณ์อาร์ซี R และอื่นๆ หลายอย่าง และคุณลักษณะ คุณสมบัติที่ต่างกันมาประยุกต์ สร้างเครื่องขยายเสียง ต่อไป

ชนิดของเครื่องขยายเสียงหรือเครื่องขยายสัญญาณเสียง

1) แอมป์หลอดสุญญากาศ (Vacum Tube) เป็นเครื่องขยายเสียงยุคแรกๆที่ใช้หลอดสุญญากาศ (Vacum Tube) ทำหน้าที่ขยายกำลังหลังของภาคขยายเสียง และยังคงมีใช้อยู่ในกลุ่มผู้หลงไหลในคุณภาพของเสียง และเอกลักษณ์ที่ดี ชัดเจน และชอบรูปแบบหลอดสุญญากาศ (Vacum Tube) แต่ข้อเสียคือกินกำลังไฟสูงมากเมื่อเทียบกับ เครื่องขยายเสียง แอมพลิฟายเออร์ (Amplifier) ชนิดอื่นๆ

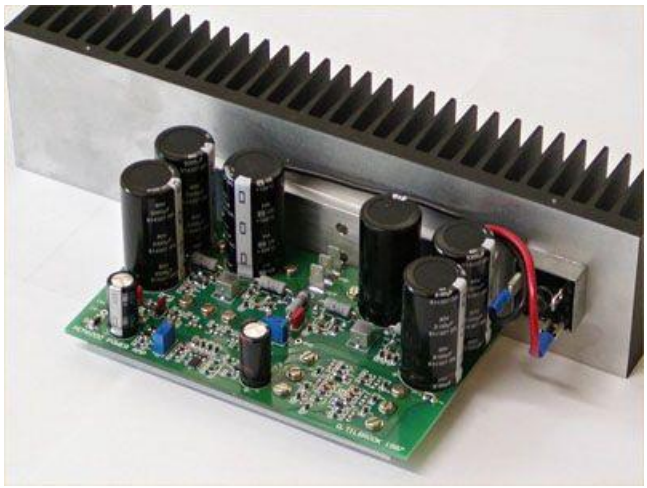
รูปภาพที่ 2.12 แอมป์หลอดสุญญากาศ
ที่มา https://www.audio2home.com/

2) แอมป์ทรานซิสเตอร์ (Transistor) เป็นเครื่องขยายเสียงที่ใช้ทรานซิสเตอร์ (Transistor) ทำหน้าที่ขยายกำลังหลักของภาคขยายเสียง ที่มีการใช้กันอย่างแพร่หลายในปัจจุบัน เนื่องจากการออกแบบวงจรได้หลากหลายรูปแบบ จนพัฒนาให้มีคุณภาพเสียงที่ดี อัตราขยายเสียงสูงความผิดเพี้ยนต่ำ สามารถหาอุปกรณ์ ซ่อมแซมได้ไม่ยาก



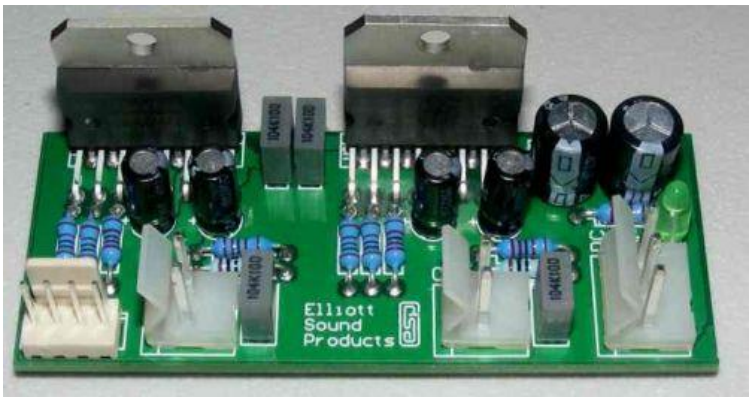
รูปภาพที่ 2.13 แอมป์ทรานซิสเตอร์
ที่มา https://www.audio2home.com/

3) แอมป์ มอสเฟส (MOSFET ,FET)เป็นเครื่องขยายเสียงที่ใช้มอสเฟส (MOSFET ,FET)ทำหน้าที่ขยายกำลังหลักของภาคขยายเสียง ในยุคหลังจาก แอมป์หลอดสูญญากาศ (Vacum Tube) ในการพัฒนาเครื่องขยายเสียงได้มีการนำสารกึ่งตัวนำมาใช้ในการทำ สร้างวงจรขยายเสียง โดยพระเอกหลักที่เป็นที่รู้จัก กันคือ ทรานซิสเตอร์ (Transistor) และมอสเฟส (MOSFET ,FET) แต่สำหรับชาว อิเล็กทรอนิกส์ หรือผู้คลั่งไคล้ หลงไหล ในการ ออกแบบ สร้าง ประดิษฐ์ วงจรเครื่องขยายเสียง จะทราบดีว่าในขณะนั้น เครื่องขยายเสียงที่ออกแบบด้วย มอสเฟส (MOSFET ,FET) ให้คุณภาพเสียงที่อิ่ม หนักแน่น มีเนื้อ กว่าการใช้ทรานซิสเตอร์ (Transistor) และราคาก็ย่อมสูงกว่าตามไปด้วย ซึ่งต่อมาตลาดของแอมป์ทรานซิสเตอร์ (Transistor)ก็กลับมาแย่งชิงตลาดกลับคืน ด้วยเทคโนโลยีการออกแบบที่ดีขึ้น แต่เครื่องขยายเสียงมอสเฟส (MOSFET ,FET) ก็ยังคงอยู่ในใจใครบางคนต่อไป



รูปภาพที่ 2.14 แอมป์มอสเฟส
ที่มา https://www.audio2home.com/

4) แอมป์ที่ใช้วงจรรวม (Integrated circuit) หรือเรียกย่อๆว่า ไอ ซี (IC.)เป็นเครื่องขยายเสียงที่ใช้วงจรรวม (Integrated circuit)ทำหน้าที่ขยายกำลังหลักของภาคขยายเสียง โดยอาศัยหลักการรวมสารกึ่งตัวนำของอุปกรณ์ ทรานซิสเตอร์ (Transistor) หรือมอสเฟส (MOSFET ,FET) บวกอุปกรณ์ R, L ,C และอื่นๆ รวมกันให้เล็กลง อยู่ บน ตัวอุปกรณ์ ตัวเดียวที่มีหลายขาและเรียกว่า (Integrated circuit) หรือเรียกย่อๆว่า ไอ ซี (IC.) นั่นเอง ข้อดีคือ มีขนาดเล็ก ความผิดเพี้ยนน้อยเมื่อใช้ในอุณหภูมิตาม Spec กำหนด ให้คุณภาพเสียงดี เหมาะกับเครื่องใช้ไฟฟ้าที่มีขนาดเล็ก ไม่เปลืองพื้นที่ ราคาถูก ข้อเสียหากมีการออกแบบอุปกรณ์ต่อพ่วง IC. ที่ไม่ดี ทำให้ความร้อนขณะใช้งานสูง และความผิดเพี้ยนสูงขึ้นได้ อีกทั้งกรณี IC. เสีย ต้องเปลี่ยน IC ซึ่งเป็นตัวหลัก ตัวเดียว ซึ่งค่า ซ่อม จะสูงมาก ถึง 70 -80 % ราคาเครื่องที่ซื้อมา ไม่คุ้มซ่อม แต่ถ้าใช้ในอุปกรณ์อื่นที่ไม่ใช่เครื่องขยายเสียง มันก็ไม่ไช่อุปกรณ์หลักเมื่อเทียบกับอุปกรณ์อื่นๆ



รูปภาพที่ 2.15 แอมป์ที่ใช้วงจรรวม
ที่มา https://www.audio2home.com/

บทที่ 3
วิธีดำเนินโครงงาน

การจัดทำโครงงาน เครื่องประกาศคำสั่งอัตโนมัติ คณะผู้จัดทำโครงงานมีวิธีดำเนินงานโครงงานตามขั้นตอน ดังนี้

3.1 แผนการดำเนินงาน

3.2 เครื่องมือ วัสดุอุปกรณ์ ในการทำโครงงาน

3.3 ขั้นตอนการทำโครงงาน

3.4 หลักการทำงานของเครื่องประกาศคำสั่งอัตโนมัติ

3.5 วงจรการทำงาน

3.1 แผนการดำเนินงาน

ตารางที่ 3.1 แผนการดำเนินงาน

ตารางการปฏิบัติงาน	ระยะเวลาในการดำเนินงาน																			
	พ.ย.65				ธ.ค.65				ม.ค.66				ก.พ.66				มี.ค.66			
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1. ศึกษาข้อมูลเพื่อหาหัวข้อโครงงานฯ																				
2. เสนอหัวข้อโครงงาน																				
2.1 เสนอหัวข้อโครงงานฯ ต่อครูที่ปรึกษาโครงงานฯ																				
2.2 จัดทำเอกสาร เสนอขออนุมัติจัดทำโครงงานฯ																				
2.3 เสนอ รร.อล.กวก.อล.ทร. ขออนุมัติจัดทำโครงงานฯ																				
3. ดำเนินการจัดทำโครงงานฯ																				
3.1 ศึกษาข้อมูลอุปกรณ์																				
3.2 ศึกษาการเขียนโค้ดภาษา C++ สำหรับ Arduino IDE																				
3.3 จัดหาอุปกรณ์																				
3.4 ทดสอบอุปกรณ์																				
3.5 วิเคราะห์ และออกแบบระบบการทำงาน																				
3.6 ทดสอบระบบ																				
3.7 ปรับปรุงแก้ไขระบบ																				
3.8 จัดทำรูปเล่ม และเอกสาร																				
3.9 วิเคราะห์ออกแบบโครงสร้างและประกอบชิ้นงาน																				
4. ส่งชิ้นงาน และเอกสารรูปเล่มโครงงานฯ																				
5. นำเสนอโครงงานฯ																				
6. จัดนิทรรศการโครงงานฯ																				

3.2 เครื่องมือ วัสดุอุปกรณ์ ในการทำโครงงาน

1) Hardware

- บอร์ด Arduino Uno R3
- Module Relay 2 Channel
- Light Emitting Diode (LED)
- Real Time Clock DS3231 (RTC)
- Push Button Switch
- LCD 20*4 I2C
- Multimeter Digital

2) Software

- ภาษา C/C++
- โปรแกรม Arduino IDE

3.3 ขั้นตอนการทำโครงงาน

1) โครงสร้างกล่องสำหรับจัดเก็บชิ้นงาน

- โครงสร้าง/กล่องทำจากแผ่นอะคริลิก
- ขนาด กว้าง x ยาว x สูง : 40 cm x 50 cm x 40 cm
- ขนาด กว้าง x ยาว : 10 cm x 40 cm

2) การเขียน และทดสอบโปรแกรม

- การเขียนโปรแกรมใช้ โปรแกรม Arduino IDE ในการเขียนโปรแกรมลงบอร์ด Arduino Uno R3 ด้วยภาษา C ,C++

3) การประกอบติดตั้งชิ้นงาน

- ในการประกอบชิ้นงานจะนำแผ่นอะคริลิกมาวัดขนาด ตัด และเจาะรูตามที่ได้ออกแบบไว้

รูปภาพที่ 3.1 ตัดแผ่นอะคริลิก

- ประกอบโครงสร้างชิ้นงาน

รูปภาพที่ 3.2 ประกอบโครงสร้าง

- นำอุปกรณ์ที่ใช้ในงานมาติดตั้ง เชื่อมสายไฟ และลงโปรแกรม

รูปภาพที่ 3.3 ติดตั้งอุปกรณ์ในโครงสร้าง

4) การทดสอบหลังติดตั้งชิ้นงาน

- ทดสอบการทำงานของอุปกรณ์ในโครงงาน รอเมื่อถึงเวลา Arduino จะดึงข้อมูลจาก RTC ไปบอร์ด Maxxtronic 115 และจะส่งเสียงให้ Amplifier ขยายแล้วส่งออกไปยังลำโพง

รูปภาพที่ 3.4 ทดสอบชิ้นงาน

5) การใช้เครื่องมือวัดทางไฟฟ้า

- Multimeter Digital

นำ Multimeter Digital มาวัดขั้วไฟ

รูปภาพที่ 3.5 การใช้ Multimeter Digital

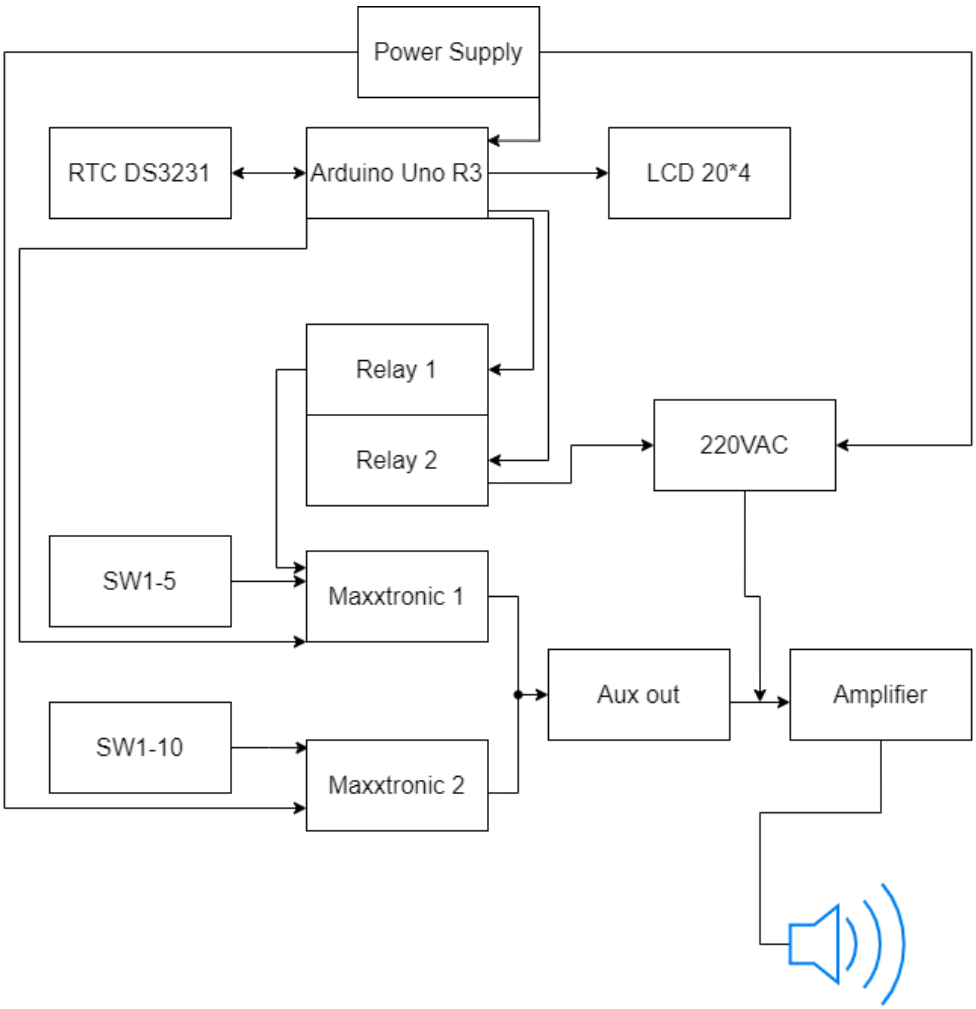
7) ปัญหาที่เกิดขึ้น

- เมื่อกดสวิตซ์ใดสวิตซ์หนึ่งในวงจรจะมีสวิตซ์ตัวอื่นมาทำงานร่วมกัน
- Module Sd Card ไม่สามารถทำงานได้ทีละเพลง

8) วิธีแก้ปัญหา

- นำไดโอดมาต่อที่ขาไฟบวกเพื่อหยุดกระแสไฟไม่ให้ไหลไปยังสวิตซ์อื่น
- เปลี่ยนจากการใช้ Module Sd Card เป็นบอร์ด Maxxtronic 115

3.4 หลักการทำงานเครื่องประกาศคำสั่งอัตโนมัติ



รูปภาพที่ 3.6 บล็อกไดอะแกรมการทำงาน

หลักการงานแบบ Automatic

นำ Adapter 5V/2A เสียบที่ปลั๊กไฟ 220V Ac และเสียบเข้าที่บอร์ด Aduino และ Arduino จะจ่ายไฟให้ส่วนต่างๆที่ใช้ไฟ 5V เลี้ยง Arduino จะเป็นตัวออกคำสั่งควบคุมการทำงานให้ตัว RTC ทำงานส่งค่าเวลามาให้ Arduino ตัว Arduino จะเขียนโค้ดให้เป็นเวลา 0900 เมื่อตัว RTC ถึงเวลาที่เขียนโค้ด Arduino จะสั่งให้บอร์ด Maxxtronic 115 ดึงเพลงที่ตั้งไว้ใน SD Card ทำงานตามเวลาที่กำหนดใช้ในบอร์ด Arduino และเมื่อตัวเสียงออกมาแล้วจะใช้ Amplifier เป็นตัวขยายเสียงให้ดังยิ่งขึ้น และบอร์ด Arduino ก็ส่งข้อมูลวัน เดือน ปี และเวลา ไปแสดงผลที่จอ LCD

* แต่มีข้อผิดพลาดที่ Sw07 ที่บอร์ด Maxxtronic 115 ไม่ทำงานตามคำสั่ง เกิดจาก Tig ส่งสัญญาณของ Arduino เกิดการจำกัดของบอร์ด Maxxtronic 115 ซึ่งทำให้ไม่สามารถใช้งานได้ จึงได้นำ Relay มาใช้เป็นสะพานตัดต่อไฟ เพื่อให้ Relay ไปสั่งการให้ Sw07 ทำงานได้ปกติ

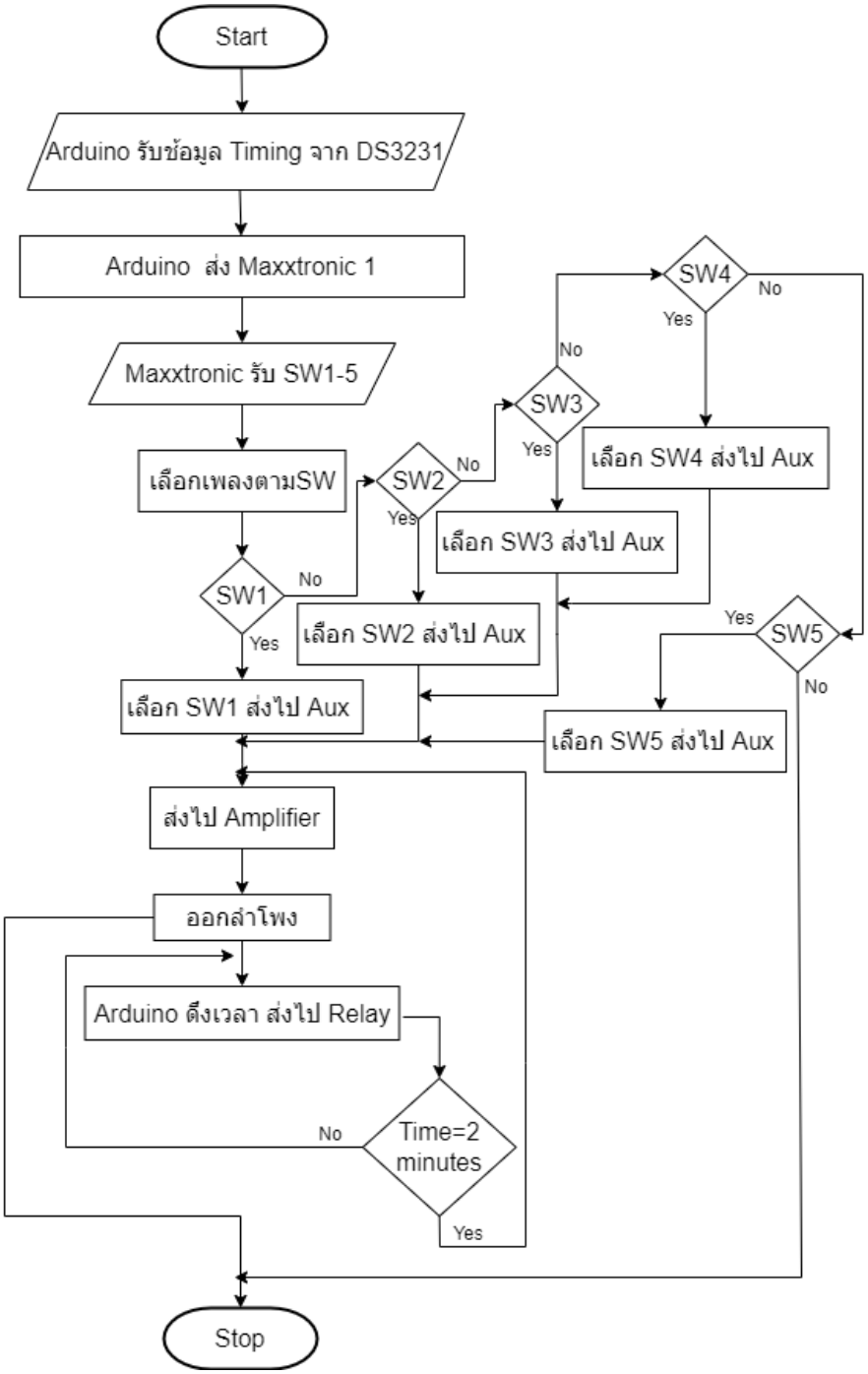
เมื่อได้กำหนดโค้ดใช้ในตัวพอร์ต Arduino ได้แล้วยังสามารถเปิด – ปิด Amplifier โดยจะเปิด Amplifier ก่อนที่ Arduino จะสั่งการ 1 นาที และปิด Amplifier หลังจากที่ประกาศเสียงแล้ว 1 นาที

หลักการทำงานแบบ Manual

นำไฟ 5V เข้าบอร์ด Maxxtronic 115 และทำการต่อสาย Sw1-10 และ Sw1-10 สามารถกดตามเสียงที่บันทึกไว้ใน SD Card เป็นนัมเบอร์ 001-010 ตาม Sw1-10 เสียงจะออกและถูกขยายโดย Amplifier แล้วส่งออกไปยังลำโพง

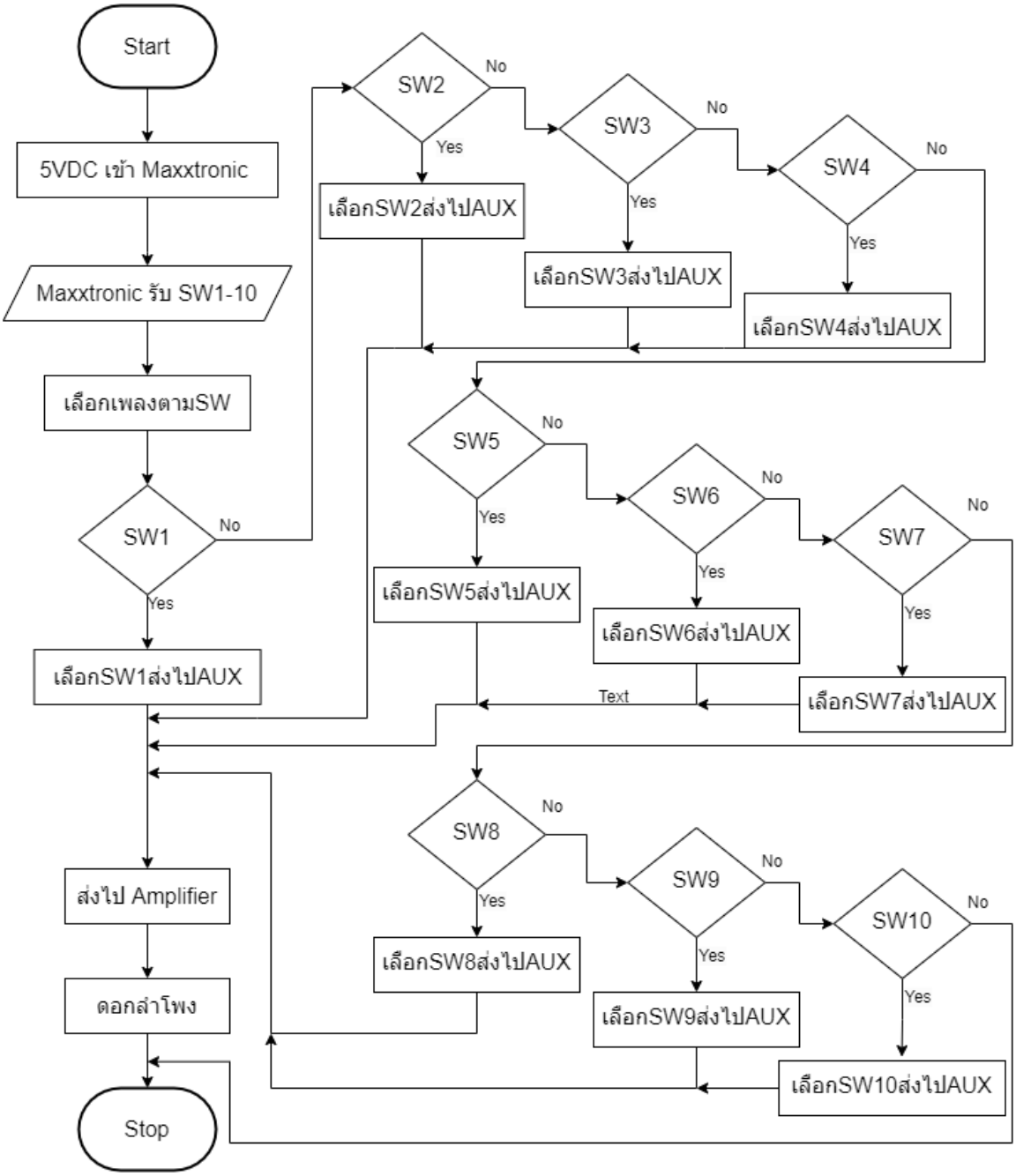
3.5 ผังการทำงานของเครื่องประกาศคำสั่งอัตโนมัติ

ผังการทำงานแบบ Automatic



รูปภาพที่ 3.7 ผังการทำงานแบบ Automatic

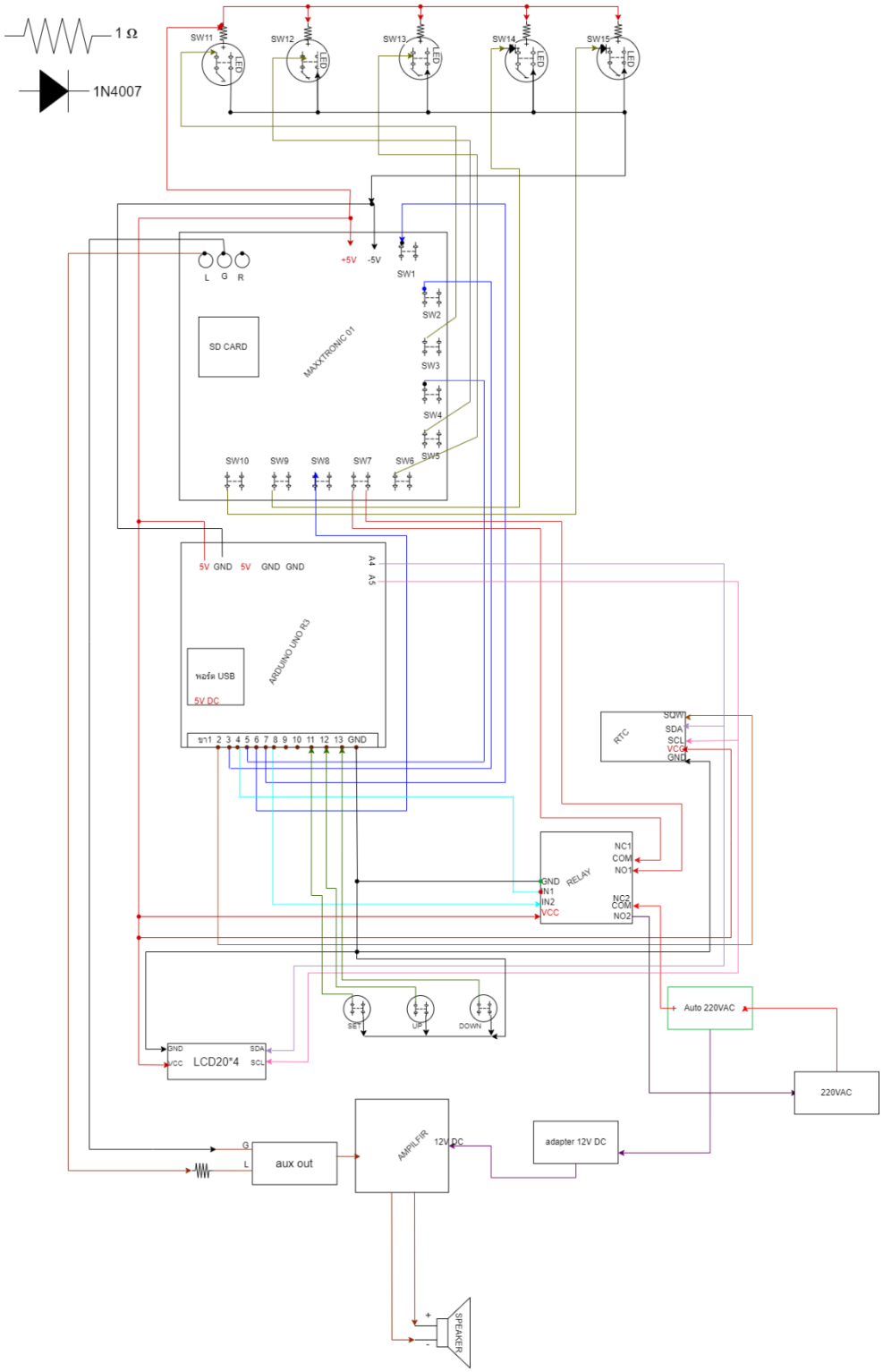
ผังการทำงานแบบ Manual



รูปภาพที่ 3.8 ผังการทำงานแบบ Manual

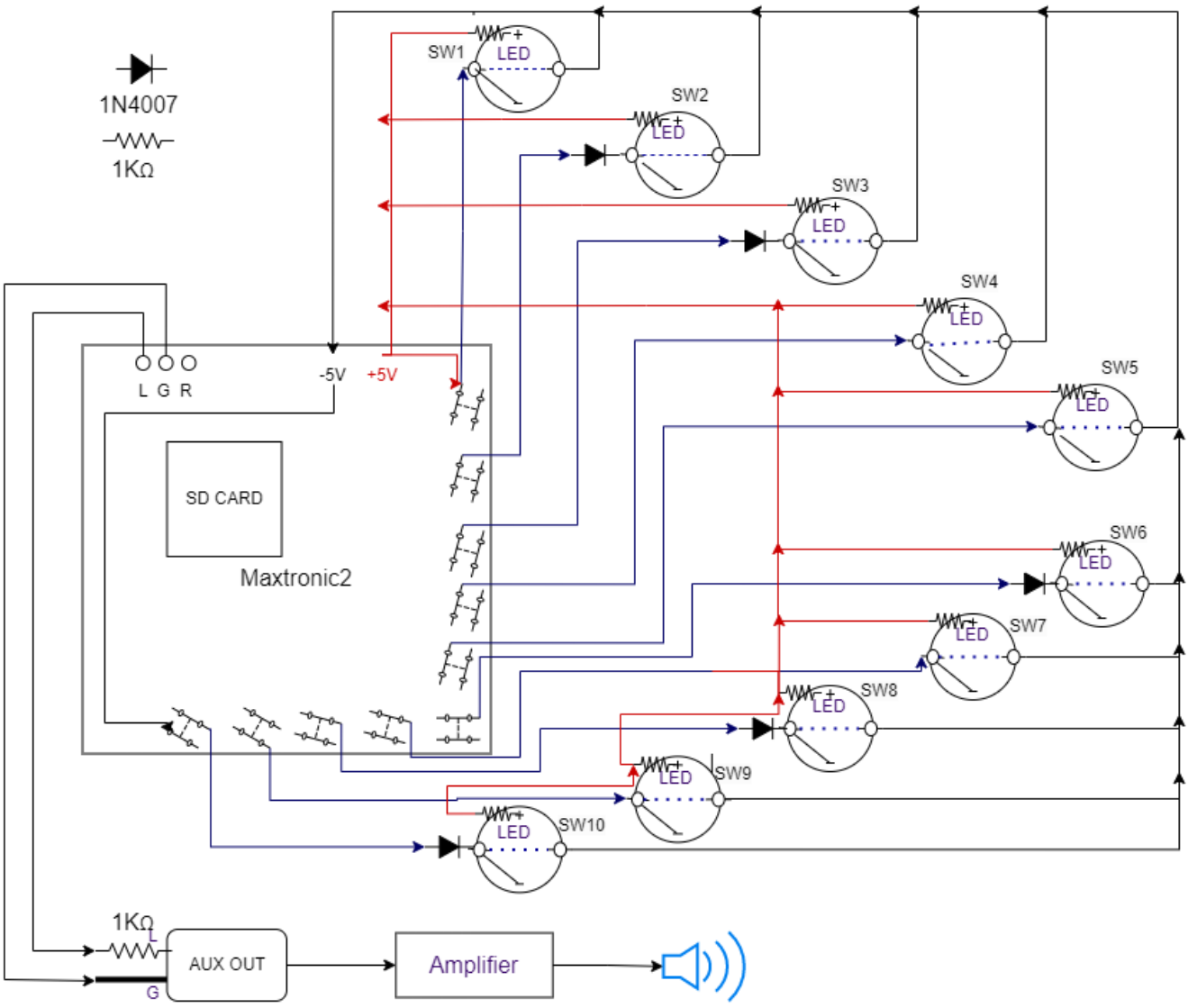
3.6 วงจรการทำงานของเครื่องประกาศคำสั่งอัตโนมัติ

วงจรการทำงานแบบ Automatic



รูปภาพที่ 3.9 วงจรการทำงานแบบ Automatic

วงจรการทำงานแบบ Manual



รูปภาพที่ 3.10 วงจรการทำงานแบบ Manual

บทที่ 4
ผลการทดลอง

4.1 ผลการทดลอง

1) ในการทดสอบการทำงานแบบ Automatic ของ เครื่องประกาศคำสั่งอัตโนมัติ โดยทดสอบจากการดังของเครื่องประกาศคำสั่งอัตโนมัติระหว่างวันที่ 6 มี.ค.66 – 10 มี.ค.66

การทดสอบความถูกต้องได้ผลการศึกษาตามตาราง ดังนี้

ตารางที่ 4.1 ผลการทดลองแบบ Automatic

เวลา	ผลการทดลองตามวัน/เดือน/ปี				
	6 มี.ค.66	7 มี.ค.66	8 มี.ค.66	9 มี.ค.66	10 มี.ค.66
0900	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง
0950	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง
1000	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง
1010	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง
1050	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง
1100	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง
1150	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง
1300	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง
1350	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง
1400	เสียงประกาศไม่ดัง	เสียงประกาศไม่ดัง	เสียงประกาศไม่ดัง	เสียงประกาศดัง	เสียงประกาศดัง
1410	เสียงประกาศไม่ดัง	เสียงประกาศไม่ดัง	เสียงประกาศไม่ดัง	เสียงประกาศดัง	เสียงประกาศดัง
1450	เสียงประกาศไม่ดัง	เสียงประกาศไม่ดัง	เสียงประกาศไม่ดัง	เสียงประกาศดัง	เสียงประกาศดัง
1500	เสียงประกาศไม่ดัง	เสียงประกาศไม่ดัง	เสียงประกาศไม่ดัง	เสียงประกาศดัง	เสียงประกาศดัง
1510	เสียงประกาศไม่ดัง	เสียงประกาศไม่ดัง	เสียงประกาศไม่ดัง	เสียงประกาศดัง	เสียงประกาศดัง
1600	เสียงประกาศไม่ดัง	เสียงประกาศไม่ดัง	เสียงประกาศไม่ดัง	เสียงประกาศดัง	เสียงประกาศดัง

จากการทดลองในช่วงแรกมีบางช่วงเวลาที่เสียงประกาศไม่ทำงานเนื่องจากขาที่บัดกรีขาบอร์ด Maxxtronic 115 ชนกัน เกิดจากความประมาทของผู้จัดทำโครงงานสิ่งประดิษฐ์

2) ในการทดสอบการทำงานแบบ Manual ของ เครื่องประกาศคำสั่งอัตโนมัติ โดยทดสอบจากการดังของเครื่องประกาศคำสั่งอัตโนมัติระหว่างวันที่ 6 มี.ค.66 – 10 มี.ค.66

การทดสอบความถูกต้องได้ผลการศึกษาตามตาราง ดังนี้

ตารางที่ 4.2 ผลการทดลองแบบ Manual

เวลา	ผลการทดลองตามวัน/เดือน/ปี				
	6 มี.ค.66	7 มี.ค.66	8 มี.ค.66	9 มี.ค.66	10 มี.ค.66
Sw01	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง
Sw02	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง
Sw03	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง
Sw04	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง
Sw05	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง
Sw06	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง
Sw07	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง
Sw08	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง
Sw09	เสียงประกาศไม่ดัง	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง
Sw10	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง
Sw11	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง
Sw12	เสียงประกาศไม่ดัง	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง
Sw13	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง
Sw14	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง
Sw15	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง	เสียงประกาศดัง

จากการทดลองในช่วงแรกมี Sw09 และ Sw12 ไม่ทำงานเนื่องจากสายสวิตซ์ที่บัดกรีกับบอร์ด Maxxtronic 115 เชื่อมต่อไม่ถึงกัน เกิดจากความประมาทของผู้จัดทำโครงงานสิ่งประดิษฐ์

ตารางที่ 4.3 ตารางเวลาที่สั่งเปิด-ปิด เพาเวอร์แอมป์

เวลาที่สั่งเปิด-ปิด	
เวลาเปิด	เวลาปิด
8:59	9:01
9:49	9:51
9:59	10:01
10:09	10:11
10:49	10:51
10:59	11:01
11:49	11:51
12:59	13:01
13:49	13:51
13:59	14:01
14:49	14:51
14:59	15:01
15:49	15:51
15:59	16:01

จากการทดลอง เวลาเปิด -ปิด Amplifier เป็นไปตามที่กำหนดทุกประการ

บทที่ 5
สรุป ปัญหา และข้อเสนอแนะ

5.1 สรุปผลการทำโครงงาน

จากผลการดำเนินงานโครงงานสิ่งประดิษฐ์เรื่อง เครื่องประกาศคำสั่งอัตโนมัติ เครื่องประกาศคำสั่งอัตโนมัติถูกใช้ทดแทนการเป่านกหวีดพักประจำชั่วโมงแบบเก่าได้อย่างดี และยังทำให้ได้ใช้เวลาพักประจำชั่วโมงได้ตรงตามเวลาที่โรงเรียนกำหนดไว้ และยังสามารถใช้ในการประจำสถานีบนเรือได้อย่างดี สามารถทำงานได้ตามฟังก์ชั่นและโปรแกรมที่ถูกเขียนขึ้น และเหมาะที่จะมาติดตั้งที่โรงเรียนอิเล็กทรอนิกส์ กองวิทยาการกรมอิเล็กทรอนิกส์ทหารเรือ และบนเรือ เพื่อความสะดวกสบาย ทั้งนี้โครงงานสำริดผลได้จากการทำงานเป็นหมู่คณะ ตลอดจนการร่วมกันคิดแก้ไขปัญหาข้อบกพร่องและพัฒนาต่อยอดจนสำเร็จ

5.2 ปัญหา

1) เมื่อกดสวิตซ์ใดสวิตซ์หนึ่งในวงจรจะมีสวิตซ์ตัวอื่นมาทำงานร่วมกัน

2) Module Sd Card ไม่สามารถทำงานได้ทีละเพลง

5.3 ข้อเสนอแนะ

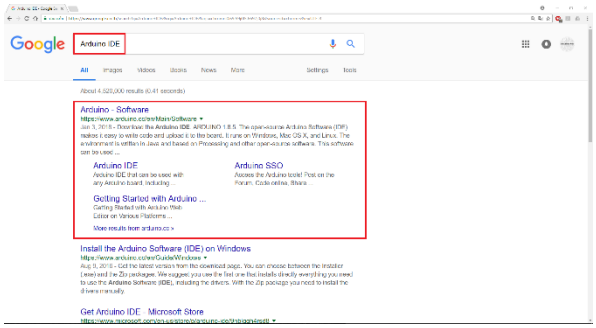
ในการจัดทำโครงงานนั้น โปรแกรม Arduino IDE นั้นมีความเข้าใจยากสำหรับผู้ดจัดทำที่มีพื้นฐานในการเขียนคำสั่งน้อย รวมไปถึงระยะเวลาที่จะศึกษาโปรแกรให้เข้าใจอย่างลึกซึ้ง ดังนั้นต้องใช้เวลามากพอสมควรเพื่อให้ผู้จัดทำสิ่งประดิษฐ์นี้ให้มีความสมบูรณ์พร้อมทุกอย่าง

ภาคผนวก

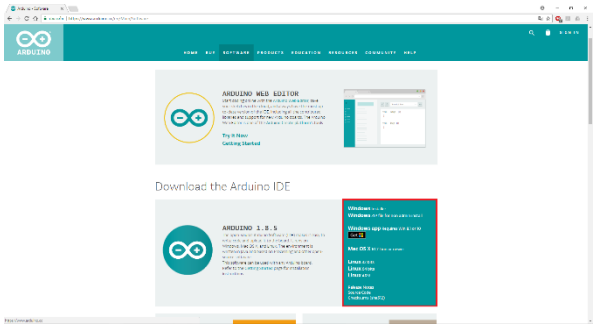
ภาคผนวก

1. ขั้นตอนการติดตั้งโปรแกรม Arduino IDE

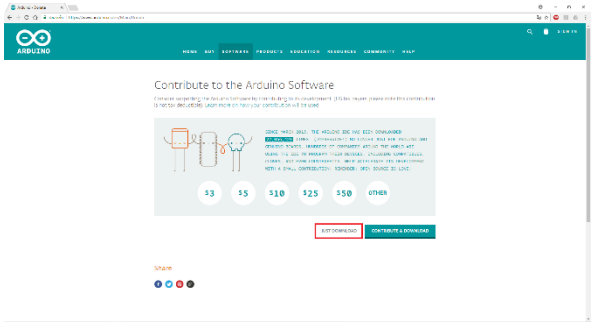
1) ค้นหาคำว่า Arduino IDE บน Google



2) จากนั้นเข้าสู่เว็บไซต์ www.arduino.cc จะปรากฏลิงค์สำหรับดาวโหลด ให้เลือกตามระบบปฏิบัติการที่ต้องการใช้งาน

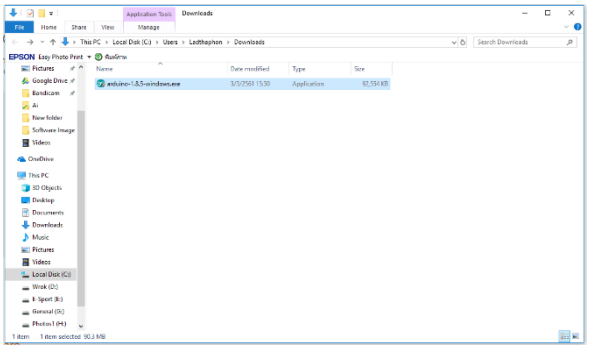


3) กดดาวน์โหลดโดยกดที่ JUST DOWNLOAD

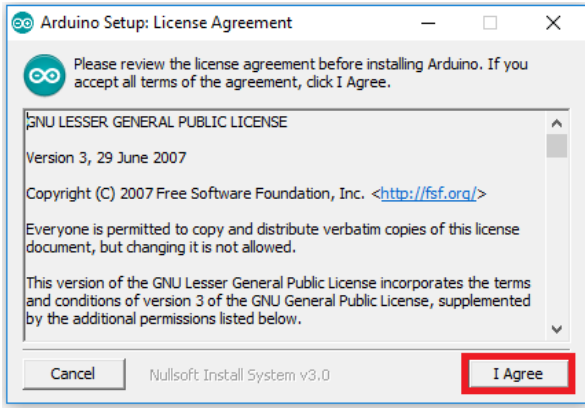


4) รอจนดาวน์โหลดเสร็จสิ้น

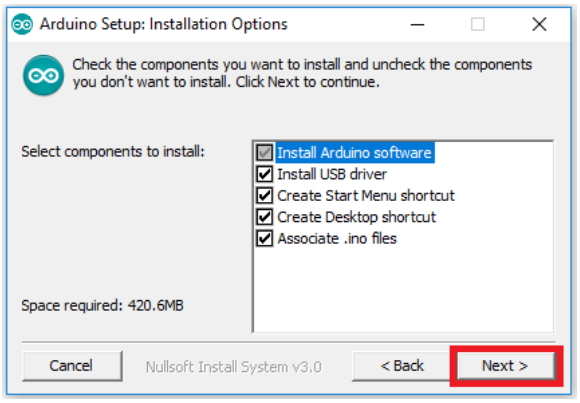
5) จะได้ไฟล์สำหรับติดตั้งโปรแกรม Arduino IDE



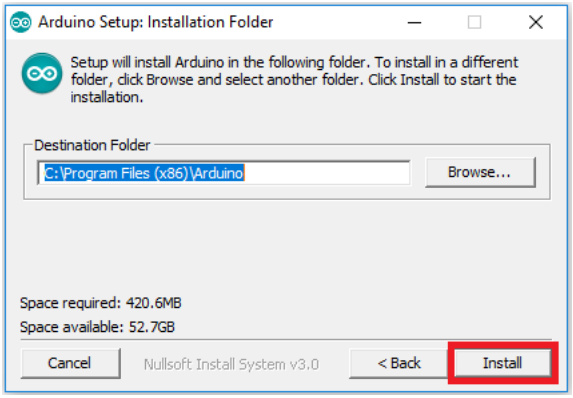
6) เปิดไฟล์ติดตั้งขึ้นมาจะปรากฏหน้าต่างสำหรับติดตั้ง กด I Agree



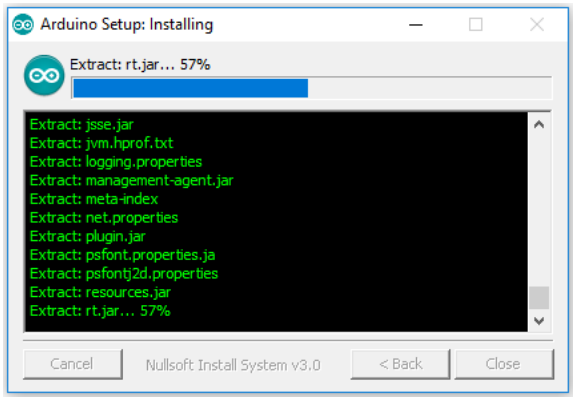
7) กด Next เพื่อไปสู่ขั้นตอนถัดไป



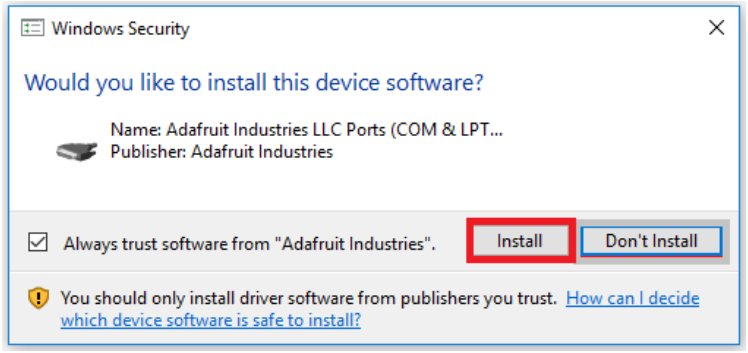
8) เลือกที่ติดตั้งโปรแกรมจากนั้น กด Install



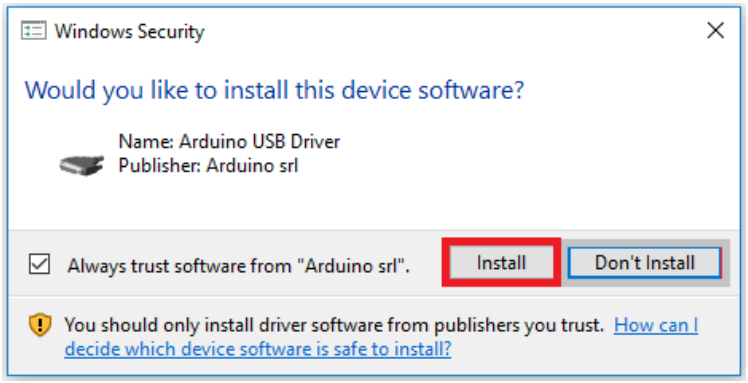
9) รอการติดตั้ง



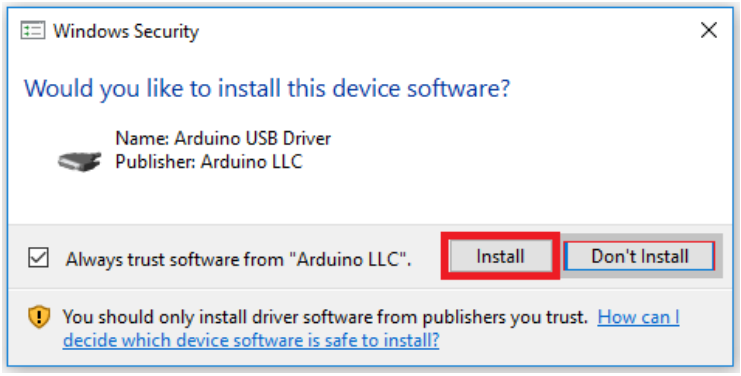
10) กด Install เพื่อติดตั้ง Adafruit Industries LLC Ports



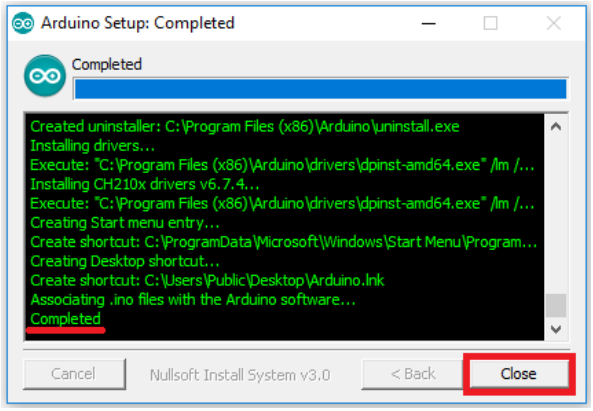
11) กด Install เพื่อติดตั้ง Arduino USB Driver srl



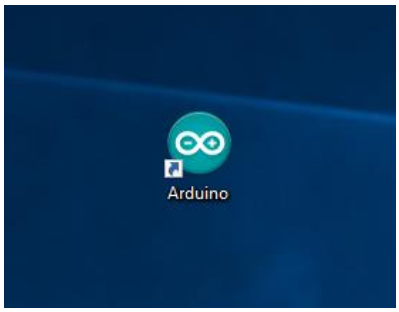
12) กด Install เพื่อติดตั้ง Arduino USB Driver LLC



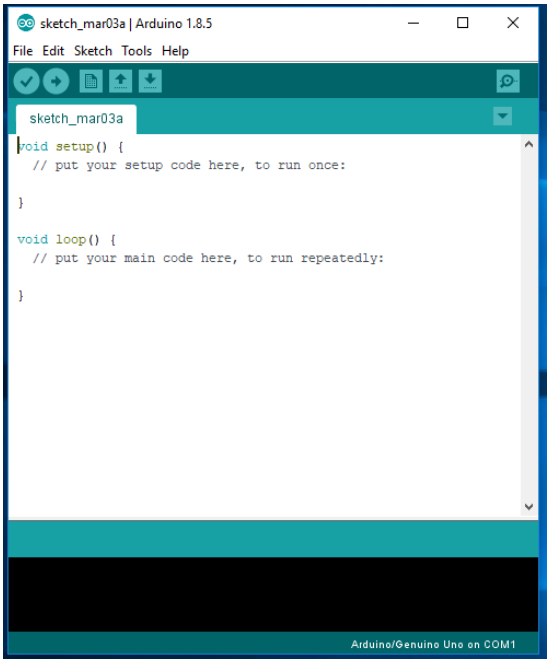
13) รอจนแสดงคำว่า Completed เป็นอันเสร็จสิ้นการติดตั้ง จากนั้นกด Close เพื่อปิด



14) จะได้ไอคอน Arduino บนหน้าจอ



15) ทดสอบเปิดโปรแกรมจะได้หน้าต่างดังรูป

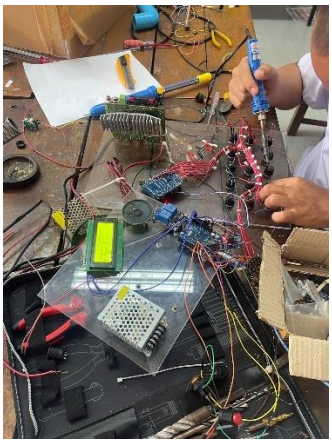


2. รูปภาพประกอบการดำเนินงาน

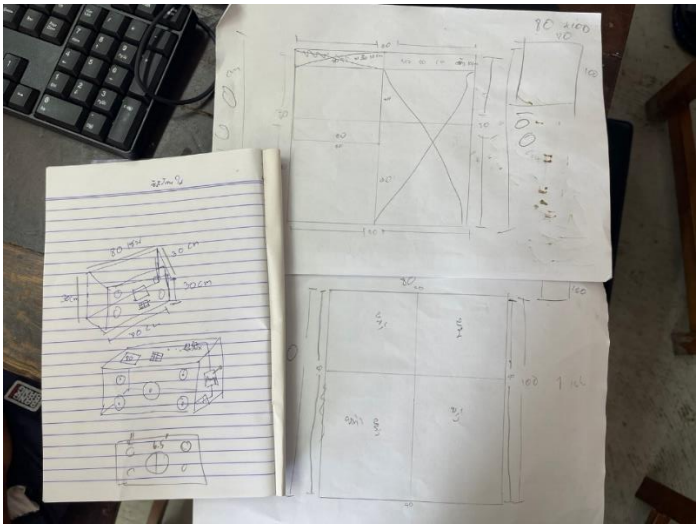
1) การใช้เครื่องมือวัด

2) งานช่าง

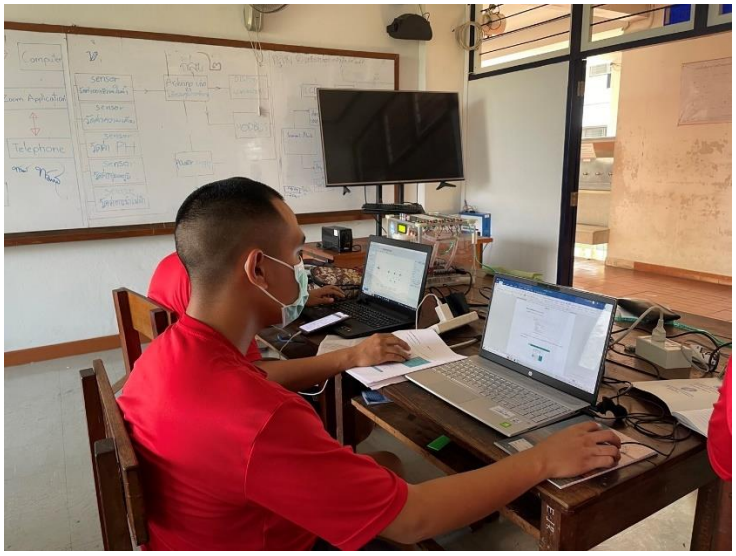
3) งานอิเล็กทรอนิกส์



4) งานออกแบบ



5) การทำโปรแกรมและเอกสาร



4. อุปกรณ์ที่ใช้ในการดำเนินงาน

ลำดับ	รายการ	จำนวน	ราคา/หน่วย	รวม (บาท)	รูปภาพ
1	Arduino Uno R3	1	270	270	
2	Maxxtronic 115	2	640	1280	
3	Relay 2 Channal	1	32	32	
4	RTC	1	280	280	
5	Push Button Switch	18	16	288	
6	LCD 20*4 I2C	1	320	320	
7	Terminal Block	1	24	24	
8	LED	4	2	8	
9	ลำโพง 6 นิ้ว	1	590	590	

5. Code ที่ใช้ทั้งหมด

1) โปรแกรมของบอร์ด Arduino Uno R3

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd(0x27, 20, 4);

#define DS3231_I2C_ADDRESS 0x68


#define   SET    11
#define   UP     12
#define   DOWN   13


byte second, minute, hour, dayOfWeek, dayOfMonth, month, year;
byte Trig=0;

int sw1    = 7;
int sw2    = 3;
int Relay1 = 4;
int sw4    = 5;
int sw8    = 6;
int Relay2 = 8;

int       FuncSET;
int       addr[8];
boolean   SET_status,UP_status,DOWN_status;
byte      SET_count,UP_count,DOWN_count;
long      Old,Last;
```

```
void setup()
{
 Serial.begin(9600);
 ds3231_1Hz();
 attachInterrupt(0, INT0_serv , RISING);

 lcd.begin();
 lcd.backlight();


 pinMode(sw1,OUTPUT);
 pinMode(sw2,OUTPUT);
 pinMode(Relay1, OUTPUT);
        digitalWrite(Relay1, LOW);
        digitalWrite(Relay1, HIGH);
 pinMode(sw4,OUTPUT);
 pinMode(sw8,OUTPUT);
  digitalWrite(sw1,1);
  digitalWrite(sw2,1);
  digitalWrite(sw4,1);
  digitalWrite(sw8,1);
 pinMode(Relay2, OUTPUT); // กำหนดขาทำหน้าที่ให้ขา 8 เป็น OUTPUT ตั้งเวลาปิดเปิดamp
        digitalWrite(Relay2, LOW);
        digitalWrite(Relay2, HIGH);


 pinMode(SET,INPUT_PULLUP);
 pinMode(UP,INPUT_PULLUP);
 pinMode(DOWN,INPUT_PULLUP);
 lcd.setCursor(0, 0); lcd.print("                    ");
 lcd.setCursor(0, 1); lcd.print("Time:  :  :    ");
 lcd.setCursor(0, 2); lcd.print("Date:  /  /    ");
```

```
 lcd.setCursor(0, 3); lcd.print("                    ");
 lcd.noCursor();
 SET_status=0;
 UP_status=0;
 DOWN_status=0;
 FuncSET=8;
}
void loop()
{
  set_Time();
  if(Trig==1)
  {
    getDateDs3231(&second, &minute, &hour, &dayOfWeek, &dayOfMonth, &month,
&year);

    addr[6]=year;
    addr[5]=month;
    addr[4]=dayOfMonth;
    addr[3]=dayOfWeek;
    addr[2]=hour;
    addr[1]=minute;
    addr[0]=second;

    display_Time_COM();
    display_Time_LCD();
    check_time();
    Trig=0;
  }
}

void check_time(void)
{
```

```
 if(dayOfWeek!=06 && dayOfWeek!=07)

{
       if(hour== 8 && minute==59 && second==00){digitalWrite(Relay2, LOW); }
  else if(hour== 9 && minute==00 && second==00){digitalWrite(sw1,0); delay(500);
digitalWrite(sw1,1);}
  else if(hour== 9 && minute==01 && second==00){digitalWrite(Relay2, HIGH);}

  else if(hour== 9 && minute==49 && second==00){digitalWrite(Relay2, LOW);}
  else if(hour== 9 && minute==50 && second==00){digitalWrite(sw2,0); delay(500);
digitalWrite(sw2,1);}
  else if(hour== 9 && minute==51 && second==00){digitalWrite(Relay2, HIGH);}

  //else if(Hour==10 && Minut==00 && second==00){digitalWrite(sw7,0); delay(500);
digitalWrite(sw7,1);}
  else if(hour== 9 && minute==59 && second==00){digitalWrite(Relay2, LOW);}
  else if(hour==10 && minute==00 && second==00){digitalWrite(Relay1, LOW);
                                          delay(1000);
                                          digitalWrite(Relay1, HIGH);
                                          delay(1000);
                                         }
  else if(hour==10 && minute==01 && second==00){digitalWrite(Relay2, HIGH);}

  else if(hour==10 && minute== 9 && second==00){digitalWrite(Relay2, LOW);}
  else if(hour==10 && minute==10 && second==00){digitalWrite(sw1,0); delay(500);
digitalWrite(sw1,1);}
  else if(hour==10 && minute==11 && second==00){digitalWrite(Relay2, HIGH);}

  else if(hour==10 && minute==49 && second==00){digitalWrite(Relay2, LOW);}
  else if(hour==10 && minute==50 && second==00){digitalWrite(sw2,0); delay(500);
digitalWrite(sw2,1);}
  else if(hour==10 && minute==51 && second==00){digitalWrite(Relay2, HIGH);}
```

```
  else if(hour==10 && minute==59 && second==00){digitalWrite(Relay2, LOW);}
  else if(hour==11 && minute==00 && second==00){digitalWrite(Relay1, LOW);
                                        delay(1000);
                                        digitalWrite(Relay1, HIGH);
                                        delay(1000);
                                       }
  else if(hour==11 && minute==01 && second==00){digitalWrite(Relay2, HIGH);}

  else if(hour==11 && minute== 9 && second==00){digitalWrite(Relay2, LOW);}
  else if(hour==11 && minute==10 && second==00){digitalWrite(sw1,0); delay(500);
digitalWrite(sw1,1);}
  else if(hour==11 && minute==11 && second==00){digitalWrite(Relay2, HIGH);}

  else if(hour==11 && minute==49 && second==00){digitalWrite(Relay2, LOW);}
  else if(hour==11 && minute==50 && second==00){digitalWrite(sw4,0); delay(500);
digitalWrite(sw4,1);}
  else if(hour==11 && minute==51 && second==00){digitalWrite(Relay2, HIGH);}

  else if(hour==12 && minute==59 && second==00){digitalWrite(Relay2, LOW);}
  else if(hour==13 && minute==00 && second==00){digitalWrite(sw1,0); delay(500);
digitalWrite(sw1,1);}
  else if(hour==13 && minute==01 && second==00){digitalWrite(Relay2, HIGH);}

  else if(hour==13 && minute==49 && second==00){digitalWrite(Relay2, LOW);}
  else if(hour==13 && minute==50 && second==00){digitalWrite(sw2,0); delay(500);
digitalWrite(sw2,1);}
  else if(hour==13 && minute==51 && second==00){digitalWrite(Relay2, HIGH);}


  else if(hour==13 && minute==59 && second==00){digitalWrite(Relay2, LOW);}
  else if(hour==14 && minute==00 && second==00){digitalWrite(Relay1, LOW);
                                        delay(1000);
```

```
                                    digitalWrite(Relay1, HIGH);
                                    delay(1000);
                                   }
  else if(hour==14 && minute==01 && second==00){digitalWrite(Relay2, HIGH);}

  else if(hour==14 && minute== 9 && second==00){digitalWrite(Relay2, LOW);}
  else if(hour==14 && minute==10 && second==00){digitalWrite(sw1,0); delay(500);
digitalWrite(sw1,1);}
  else if(hour==14 && minute==11 && second==00){digitalWrite(Relay2, HIGH);}

  else if(hour==14 && minute==49 && second==00){digitalWrite(Relay2, LOW);}
  else if(hour==14 && minute==50 && second==00){digitalWrite(sw2,0); delay(500);
digitalWrite(sw2,1);}
  else if(hour==14 && minute==51 && second==00){digitalWrite(Relay2, HIGH);}

  else if(hour==14 && minute==59 && second==00){digitalWrite(Relay2, LOW);}
  else if(hour==15 && minute==00 && second==00){digitalWrite(Relay1, LOW);
                                    delay(1000);
                                    digitalWrite(Relay1, HIGH);
                                    delay(1000);
                                   }
  else if(hour==15 && minute==01 && second==00){digitalWrite(Relay2, HIGH);}

  else if(hour==15 && minute== 9 && second==00){digitalWrite(Relay2, LOW);}
  else if(hour==15 && minute==10 && second==00){digitalWrite(sw1,0); delay(500);
digitalWrite(sw1,1);}
  else if(hour==15 && minute==01 && second==00){digitalWrite(Relay2, HIGH);}

  else if(hour==15 && minute==59 && second==00){digitalWrite(Relay2, LOW);}
  else if(hour==16 && minute==00 && second==00){digitalWrite(sw8,0); delay(500);
digitalWrite(sw8,1);}
  else if(hour==16 && minute==01 && second==00){digitalWrite(Relay2, HIGH);}
```

```
}
}

void INT0_serv(void)
{
  Trig=1;
}

void ds3231_1Hz(void)
{
 Wire.begin();
 Wire.beginTransmission(DS3231_I2C_ADDRESS);
 Wire.write(0x0E);
 Wire.write(0x00);
 Wire.endTransmission();
}

byte decToBcd(byte val)
{
  return ( (val/10*16) + (val%10) );
}

byte bcdToDec(byte val)
{
  return ( (val/16*10) + (val%16) );
}

void stopDs3231()
{
  Wire.beginTransmission(DS3231_I2C_ADDRESS);
  Wire.write(0);
  Wire.write(0x80);
```

```
  Wire.endTransmission(0);
}

void setDateDs3231(byte second,byte minute,byte hour,byte dayOfWeek,byte
dayOfMonth,byte month,byte year)          // 0-99
{
   Wire.beginTransmission(DS3231_I2C_ADDRESS);
   Wire.write(0);
   Wire.write(decToBcd(second));    // 0 to bit 7 starts the clock
   Wire.write(decToBcd(minute));
   Wire.write(decToBcd(hour));      // If you want 12 hour am/pm you need to set
                                    // bit 6 (also need to change readDateDs1307)
   Wire.write(decToBcd(dayOfWeek));
   Wire.write(decToBcd(dayOfMonth));
   Wire.write(decToBcd(month));
   Wire.write(decToBcd(year));
   Wire.endTransmission();
}

void getDateDs3231( byte *second,byte *minute, byte *hour,byte *dayOfWeek,byte
*dayOfMonth,byte *month,byte *year)
{
  // Reset the register pointer
  Wire.beginTransmission(DS3231_I2C_ADDRESS);
  Wire.write(0);
  Wire.endTransmission();
  Wire.requestFrom(DS3231_I2C_ADDRESS, 7);


  *second     = bcdToDec(Wire.read());  //& 0x7f
  *minute     = bcdToDec(Wire.read());
  *hour       = bcdToDec(Wire.read());  //& 0x3f Need to change this if 12 hour am/pm
```

```
  *dayOfWeek  = bcdToDec(Wire.read());
  *dayOfMonth = bcdToDec(Wire.read());
  *month      = bcdToDec(Wire.read());
  *year       = bcdToDec(Wire.read());
}

void display_Time_COM(void)
{
  getDateDs3231(&second, &minute, &hour, &dayOfWeek, &dayOfMonth, &month, &year);
  if(hour<10){Serial.print(0);}Serial.print(hour, DEC);              Serial.print(":");
  if(minute<10){Serial.print(0);}Serial.print(minute, DEC);          Serial.print(":");
  if(second<10){Serial.print(0);}Serial.print(second, DEC);          Serial.print("   ");
  if(dayOfMonth<10){Serial.print(0);}Serial.print(dayOfMonth, DEC); Serial.print("/");
  if(month<10){Serial.print(0);}Serial.print(month, DEC);            Serial.print("/");
  if(year<10){Serial.print(0);}Serial.println(year, DEC);

}

void display_Time_LCD(void)
{
  lcd.setCursor(5, 1);if(hour  <10){lcd.print(0);}lcd.print(hour, DEC);
  lcd.setCursor(8, 1);if(minute<10){lcd.print(0);}lcd.print(minute, DEC);
  lcd.setCursor(11,1);if(second<10){lcd.print(0);}lcd.print(second, DEC);
  lcd.setCursor(15,0);if(dayOfWeek<10){lcd.print(0);}lcd.print(dayOfWeek, DEC);

      lcd.setCursor( 7, 0);
      if(dayOfWeek==1){lcd.print("MONDAY");}
      lcd.setCursor( 6, 0);
      if(dayOfWeek==2){lcd.print("TUESDAY");}
      lcd.setCursor( 5, 0);
      if(dayOfWeek==3){lcd.print("WEDNESDAY");}
      lcd.setCursor( 6, 0);
```

```
      if(dayOfWeek==4){lcd.print("THURSDAY");}
      lcd.setCursor( 7, 0);
      if(dayOfWeek==5){lcd.print("FRIDAY");}
      lcd.setCursor( 6, 0);
      if(dayOfWeek==6){lcd.print("SATURDAY");}
      lcd.setCursor( 7, 0);
      if(dayOfWeek==7){lcd.print("SUNDAY");}

  lcd.setCursor(5, 2);if(dayOfMonth<10){lcd.print(0);}lcd.print(dayOfMonth, DEC);
  lcd.setCursor(8, 2);if(month <10){lcd.print(0);}lcd.print(month, DEC);
  lcd.setCursor(11,2);if(year <10){lcd.print(0);}lcd.print(year, DEC);
}


/*****************************  program set ds3231 ************************************/
void set_Time(void)
{
  Last=millis();
  if(Last - Old >5)
  {
     Old=Last;
     if(digitalRead(SET)==1 && SET_status==1)
     {
       SET_count++;
       if(SET_count>2){SET_count=0;SET_status=0;}
     }
     if(digitalRead(UP)==1 && UP_status==1)
     {
       UP_count++;
       if(UP_count>2){UP_count=0;UP_status=0;}
     }
     if(digitalRead(DOWN)==1 && DOWN_status==1)
```

```
    {
      DOWN_count++;
      if(DOWN_count>2){DOWN_count=0;DOWN_status=0;}
    }
  }
  if(digitalRead(SET)==0 && SET_status==0)
  {
    SET_status=1;
    detachInterrupt(0);
    FuncSET++;
    if(FuncSET==1)
    {
      detachInterrupt(0);
      getDateDs3231(&second, &minute, &hour, &dayOfWeek, &dayOfMonth, &month,
&year);
      addr[6]=year;
      addr[5]=month;
      addr[4]=dayOfMonth;
      addr[3]=dayOfWeek;
      addr[2]=hour;
      addr[1]=minute;
      addr[0]=second;
    }
        if(FuncSET==1){lcd.setCursor( 6, 1);lcd.cursor();}//lcd.blink();}
    else if(FuncSET==2){lcd.setCursor( 9, 1);lcd.cursor();}//lcd.blink();}
    else if(FuncSET==3){lcd.setCursor(12, 1);lcd.cursor();}//lcd.blink();}
    else if(FuncSET==4){lcd.setCursor(20, 0);lcd.cursor();}//lcd.blink();}
    else if(FuncSET==5){lcd.setCursor( 6, 2);lcd.cursor();}//lcd.blink();}
    else if(FuncSET==6){lcd.setCursor( 9, 2);lcd.cursor();}//lcd.blink();}
    else if(FuncSET==7){lcd.setCursor(12, 2);lcd.cursor();}//lcd.blink();}
    else if(FuncSET>=8)
    {
```

```
      setDateDs3231(addr[0],addr[1],addr[2],addr[3],addr[4],addr[5],addr[6]);
//sec,min,hour,dayOfweek,dayOfmonth,month,year
      FuncSET=0;lcd.noCursor();Time_display();
      attachInterrupt(0, INT0_serv , RISING);

    }
  }
  if(digitalRead(UP)==0 && UP_status==0)  /*********************UP************************/
  {
    UP_status=1;
    if(FuncSET==7)  //year 1
    {
      addr[6]++; if(addr[6]>99){addr[6]=0;}
      lcd.setCursor( 11, 2);
      if(addr[6]<10){lcd.print(0);}lcd.print(addr[6]);
      lcd.setCursor( 12, 2);lcd.cursor();
    }
    else if(FuncSET==6)  //mount 2
    {
      addr[5]++; if(addr[5]>12){addr[5]=1;}
      lcd.setCursor( 8, 2);
      if(addr[5]<10){lcd.print(0);}lcd.print(addr[5]);
      lcd.setCursor( 9, 2);lcd.cursor();
    }
    else if(FuncSET==5) //date of month 3
    {
      addr[4]++; if(addr[4]>31){addr[4]=1;}
      lcd.setCursor( 5, 2);
      if(addr[4]<10){lcd.print(0);}lcd.print(addr[4]);
      lcd.setCursor( 6, 2);lcd.cursor();
    }
    else if(FuncSET==4)  //day of week
```

```
{
  addr[3]++; if(addr[3]>7){addr[3]=1;}

  lcd.setCursor( 7, 0);
  if(addr[3]==1){lcd.print("MONDAY");}
  lcd.setCursor( 6, 0);
  if(addr[3]==2){lcd.print("TUESDAY");}
  lcd.setCursor( 5, 0);
  if(addr[3]==3){lcd.print("WEDNESDAY");}
  lcd.setCursor( 6, 0);
  if(addr[3]==4){lcd.print("THURSDAY");}
  lcd.setCursor( 7, 0);
  if(addr[3]==5){lcd.print("FRIDAY");}
  lcd.setCursor( 6, 0);
  if(addr[3]==6){lcd.print("SATURDAY");}
  lcd.setCursor( 7, 0);
  if(addr[3]==7){lcd.print("SUNDAY");}

  lcd.setCursor( 15, 0);lcd.cursor();

}
else if(FuncSET==1)  //hour 5
{
  addr[2]++; if(addr[2]>23){addr[2]=0;}
  lcd.setCursor( 5, 1);
  if(addr[2]<10){lcd.print(0);}lcd.print(addr[2]);
  lcd.setCursor( 6, 1);lcd.cursor();
}
else if(FuncSET==2)  //minute 6
{
  addr[1]++; if(addr[1]>59){addr[1]=0;}
  lcd.setCursor( 8, 1);
```

```
      if(addr[1]<10){lcd.print(0);}lcd.print(addr[1]);
      lcd.setCursor( 9, 1);lcd.cursor();
    }
    else if(FuncSET==3)  //sec 7
    {
      addr[0]++; if(addr[0]>59){addr[0]=0;}
      lcd.setCursor( 11, 1);
      if(addr[0]<10){lcd.print(0);}lcd.print(addr[0]);
      lcd.setCursor( 12, 1);lcd.cursor();
    }
  }
  if(digitalRead(DOWN)==0 &&
DOWN_status==0)/*************************DOWN*************************/
  {
    DOWN_status=1;
    if(FuncSET==7)  //year 1
    {
      addr[6]--; if(addr[6]<0){addr[6]=99;}
      lcd.setCursor( 11, 2);
      if(addr[6]<10){lcd.print(0);}lcd.print(addr[6]);
      lcd.setCursor( 12, 2);lcd.cursor();
    }
    else if(FuncSET==6)  //month 2
    {
      addr[5]--; if(addr[5]<1){addr[5]=12;}
      lcd.setCursor( 8, 2);
      if(addr[5]<10){lcd.print(0);}lcd.print(addr[5]);
      lcd.setCursor( 9, 2);lcd.cursor();
    }
    else if(FuncSET==5)  //date of month 3
    {
      addr[4]--; if(addr[4]<1){addr[4]=31;}
```

```
  lcd.setCursor( 5, 2);
  if(addr[4]<10){lcd.print(0);}lcd.print(addr[4]);
  lcd.setCursor( 6, 2);lcd.cursor();
}
else if(FuncSET==4)  //day of week
{
  addr[3]--; if(addr[3]<1){addr[3]=7;}
  lcd.setCursor( 7, 0);
  if(addr[3]==1){lcd.print("MONDAY");}
  lcd.setCursor( 6, 0);
  if(addr[3]==2){lcd.print("TUESDAY");}
  lcd.setCursor( 5, 0);
  if(addr[3]==3){lcd.print("WEDNESDAY");}
  lcd.setCursor( 6, 0);
  if(addr[3]==4){lcd.print("THURSDAY");}
  lcd.setCursor( 7, 0);
  if(addr[3]==5){lcd.print("FRIDAY");}
  lcd.setCursor( 6, 0);
  if(addr[3]==6){lcd.print("SATURDAY");}
  lcd.setCursor( 7, 0);
  if(addr[3]==7){lcd.print("SUNDAY");}

  lcd.setCursor( 15, 0);lcd.cursor();

}
else if(FuncSET==1)  //hour 5
{
  addr[2]--; if(addr[2]<0){addr[2]=23;}
  lcd.setCursor( 5, 1);
  if(addr[2]<10){lcd.print(0);}lcd.print(addr[2]);
  lcd.setCursor( 6, 1);lcd.cursor();
}
```

```
    else if(FuncSET==2)  //minute 6
    {
      addr[1]--; if(addr[1]<0){addr[1]=59;}
      lcd.setCursor( 8, 1);
      if(addr[1]<10){lcd.print(0);}lcd.print(addr[1]);
      lcd.setCursor( 9, 1);lcd.cursor();
    }
    else if(FuncSET==3) //sec 7
    {
      addr[0]--; if(addr[0]<0){addr[0]=59;}
      lcd.setCursor( 11, 1);
      if(addr[0]<10){lcd.print(0);}lcd.print(addr[0]);
      lcd.setCursor( 12, 1);lcd.cursor();
    }
  }
}

void  Time_display(void)
{
  lcd.setCursor(0, 0); lcd.print("                    ");
  lcd.setCursor(0, 1); lcd.print("Time:  :  :      ");
  lcd.setCursor(0, 2); lcd.print("Date:  /  /      ");
  lcd.setCursor(0, 3); lcd.print("                    ");
}
```

บรรณานุกรม

ข้อมูลเกี่ยวกับ *ทฤษฎีเสียง.* [ออนไลน์].
เข้าถึงได้จาก : http://medi.moph.go.th/km/2560/rsroom/rsroomp2.pdf.
(วันที่ค้นหาข้อมูล : 30 มกราคม 2566).

ข้อมูลเกี่ยวกับ *Arduino Uno R3.* [ออนไลน์].
เข้าถึงได้จาก : https://www.futurekit.com/th/content/10792/.
(วันที่ค้นหาข้อมูล : 30 มกราคม 2566).

ข้อมูลเกี่ยวกับ *Arduino IDE.* [ออนไลน์].
เข้าถึงได้จาก : https://www.futurekit.com/th/content/10793/.
(วันที่ค้นหาข้อมูล : 30 มกราคม 2566).

ข้อมูลเกี่ยวกับ *Real Time Clock DS3231 (RTC).* [ออนไลน์].
เข้าถึงได้จาก : https://www.cybertice.com/product/21/ds3231-module.
(วันที่ค้นหาข้อมูล : 30 มกราคม 2566).

ข้อมูลเกี่ยวกับ *Mxxtronic 115.* [ออนไลน์].
เข้าถึงได้จาก : https://www.futurekit.com/th/product/33325-25463/mxa115.
(วันที่ค้นหาข้อมูล : 30 มกราคม 2566).

ข้อมูลเกี่ยวกับ *Push Button Switch.* [ออนไลน์].
เข้าถึงได้จาก : https://mall.factomart.com/what-is-a-push-button-switch.
(วันที่ค้นหาข้อมูล : 30 มกราคม 2566).

ข้อมูลเกี่ยวกับ *LCD 20*4 I2C.* [ออนไลน์].
เข้าถึงได้จาก : http://www.ett.co.th/prod2018/LCD%2020%20x%204%20I2C/man-th-LCD%2020x4%20I2C.pdf
(วันที่ค้นหาข้อมูล : 30 มกราคม 2566).

คณะผู้จัดทำ

นรจ.นราวิทย์ ชูราษฎร์

ประวัติการศึกษา : ประกาศนียบัตรชั้นสูงปีที่ 1

วิทยาลัยเทคนิคประจวบคีรีขันธ์

จ. ประจวบคีรีขันธ์

Facebook : Narawit Churat

นรจ.สิทธิโชค แก้วสระแสน

ประวัติการศึกษา : มัธยมศึกษาปีที่ 6

โรงเรียนกรรณสูตศึกษาลัย

จ. สุพรรณบุรี

Facebook : สิทธิโชค แก้วสระแสน

นรจ.วรภัทร สิงหะ

ประวัติการศึกษา : มัธยมศึกษาปีที่ 6

โรงเรียนพิบูลวิทยาลัย

จ. ลพบุรี

Facebook : Worapat Singha

นรจ.ศุภพฏฐ์ บุญซ้อน

ประวัติการศึกษา : มัธยมศึกษาปีที่ 6

โรงเรียนโกวิทธำรงเชียงใหม่

จ. เชียงใหม่

Facebook : Suphaphod Boonsorn

นรจ.ปฏิวัติ นอกไธสง

ประวัติการศึกษา : มัธยมศึกษาปีที่ 6

โรงเรียนนาโพธิ์พิทยาคม

จ. บุรีรัมย์

Facebook : Patiwat Nokthaisong

เครื่องประกาศคำสั่งอัตโนมัติ
(Automatic announcement machine)

บทคัดย่อ

การจัดทำโครงงานสิ่งประดิษฐ์ เครื่องประกาศคำสั่งอัตโนมัติ เป็นการศึกษาค้นคว้าเกี่ยวกับการนำความรู้จากวิชาไมโครคอนโทรลเลอร์มาประยุกต์ใช้งานในเป่านกหวีดพักประจำชั่วโมงของ นรจ.รร.อล.กวก.อล.ทร. และสามารถประยุกต์ใช้ในการประจำสถานีบนเรือ ซึ่งเมื่อถึงเวลาสามารถประกาศคำสั่งออกได้เองโดยไม่ต้องใช้เวรยามในการประกาศ และยังสามารถกดสวิตซ์เพื่อประกาศประจำสถานีภายในเรือได้ รวมถึงโครงงานนี้ยังช่วยลดการเสียเวลาเรียนของเวรยามประจำวัน และความล่าช้าจากการประกาศคำสั่งประจำสถานีบนเรือ

ที่มาและความสำคัญ

ในปัจจุบัน รร.อล.กวก.อล.ทร. ได้มีการพักประจำชั่วโมงของ นรจ.รร.อล.กวก.อล.ทร ซึ่งต้องใช้เวรยามที่รับผิดชอบประจำวันทำหน้าที่ประกาศพักประจำชั่วโมง ซึ่งต้องใช้เวรยามในแต่ละวันมาประกาศพักประจำชั่วโมง ทำให้เวรยามประจำวันเรียนไม่ต่อเนื่องเพราะต้องขออนุญาตครูผู้สอนมาทำหน้าที่ พร้อมทั้งปัจจุบันบนเรือได้มีการประจำสถานีต่างๆ ซึ่งต้องใช้คนที่มีหน้าที่ประกาศเป็นคนต่อไมค์ประกาศ จึงอาจเกิดความล่าช้าในการประจำสถานีต่างๆ

วัตถุประสงค์ของโครงงาน

- เพื่อสร้างเครื่องประกาศคำสั่งอัตโนมัติ

ประโยชน์ที่จะได้รับ

1. เพื่อไม่ให้เสียเวลาเรียนของเวรยามประจำวัน
2. เพื่อเพิ่มความสะดวกรวดเร็วในการประจำสถายีภายในเรือ

สิ่งที่จะได้รับ

1. ได้เครื่องประกาศคำสั่งอัตโนมัติ
2. สามารถนำเครื่องประกาศคำสั่งอัตโนมัติไปใช้ใน รร.อล.กวก.อล.ทร.
3. ได้เครื่องอำนวยความสะดวกมาใช้ภายในเรือ

หลักการทำงานของเครื่องประกาศคำสั่งอัตโนมัติ

<u>การทำงานแบบ Automatic</u>

นำ Adapter 5V/2A เสียบที่ปลั๊กไฟ 220V Ac และเสียบเข้าที่บอร์ด Aduino และ Arduino จะจ่ายไฟให้ส่วนต่างๆที่ใช้ไฟ 5V เลี้ยง Arduino จะเป็นตัวออกคำสั่งควบคุมการทำงานให้ตัว RTC ทำงานส่งค่าเวลามาให้ Arduino ตัว Arduino จะเขียนโค้ดให้เป็นเวลา 0900 เมื่อตัว RTC ถึงเวลาที่เขียนโค้ด Arduino จะสั่งให้บอร์ด Maxxtronic 115 ดึงเพลงที่ตั้งไว้ใน SD Card ทำงานตามเวลาที่กำหนดใช้ในบอร์ด Arduino และเมื่อตัวเสียงออกมาแล้วจะใช้ Amplifier เป็นตัวขยายเสียงให้ดังยิ่งขึ้น และบอร์ด Arduino ก็ส่งข้อมูลวัน เดือน ปี และเวลา ไปแสดงผลที่จอ LCD

* แต่มีข้อผิดพลาดที่ Sw07 ที่บอร์ด Maxxtronic 115 ไม่ทำงานตามคำสั่ง เกิดจาก Tig ส่งสัญญาณของ Arduino เกิดการจำกัดของบอร์ด Maxxtronic 115 ซึ่งทำให้ไม่สามารถใช้งานได้ จึงได้นำ Relay มาใช้เป็นสะพานตัดต่อไฟ เพื่อให้ Relay ไปสั่งการให้ Sw07 ทำงานได้ปกติ

เมื่อได้กำหนดโค้ดใช้ในตัวพอร์ต Arduino ได้แล้วยังสามารถเปิด – ปิด Amplifier โดยจะเปิด Amplifier ก่อนที่ Arduino จะสั่งการ 1 นาที และปิด Amplifier หลังจากที่ประกาศเสียงแล้ว 1 นาที

<u>การทำงานแบบ Manual</u>

นำไฟ 5V เข้าบอร์ด Maxxtronic 115 และทำการต่อสาย Sw1-10 และ Sw1-10 สามารถกดตามเสียงที่บันทึกไว้ใน SD Card เป็นนัมเบอร์ 001-010 ตาม Sw1-10 เสียงจะออกและถูกขยายโดย Amplifier แล้วส่งออกไปยังลำโพง

สรุปผลการทดลอง

จากผลการดำเนินงานโครงงานสิ่งประดิษฐ์เรื่อง เครื่องประกาศคำสั่งอัตโนมัติ เครื่องประกาศคำสั่งอัตโนมัติถูกใช้ทดแทนการเป่านกหวีดพักประจำชั่วโมงแบบเก่าได้อย่างดี และยังทำให้ได้ใช้เวลาพักประจำชั่วโมงได้ตรงตามเวลาที่โรงเรียนกำหนดไว้ และยังสามารถใช้ในการประจำสถานีบนเรือได้อย่างดี สามารถทำงานได้ตามฟังก์ชั่นและโปรแกรมที่ถูกเขียนขึ้น และเหมาะที่จะมาติดตั้งที่โรงเรียนอิเล็กทรอนิกส์ กองวิทยาการกรมอิเล็กทรอนิกส์ทหารเรือ และบนเรือ เพื่อความสะดวกสบาย ทั้งนี้โครงงานสำริดผลได้ จากการทำงานเป็นหมู่คณะ ตลอดจนการร่วมกันคิดแก้ไขปัญหาข้อบกพร่องและพัฒนาต่อยอดจนสำเร็จ

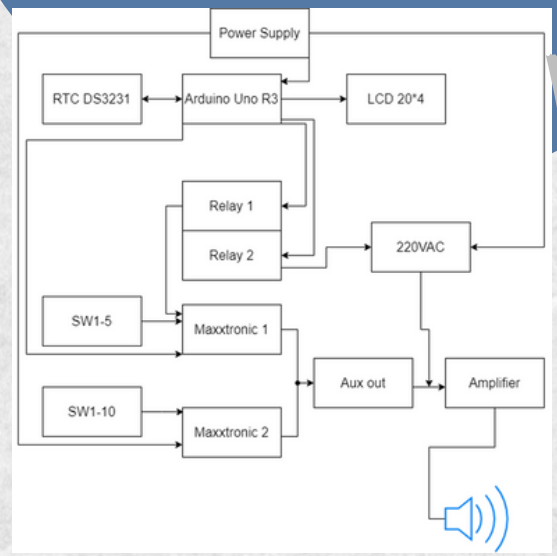
ปัญหา

1. เมื่อกดสวิตซ์ใดสวิตซ์หนึ่งในวงจรจะมีสวิตซ์ตัวอื่นมาทำงานร่วมกัน
2. Module Sd Card ไม่สามารถทำงานได้ทีละเพลง

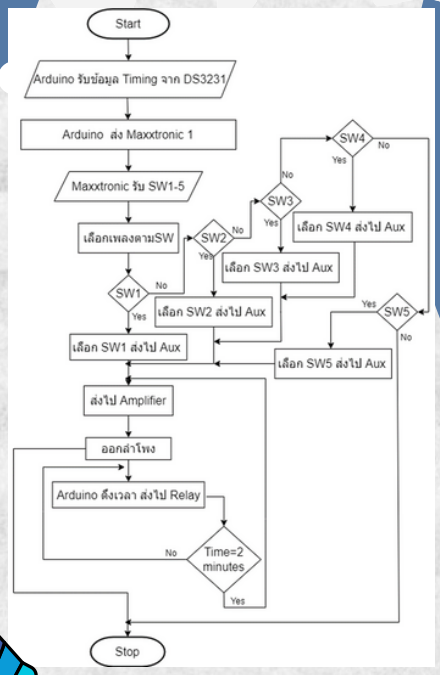
ข้อเสนอแนะ

ในการจัดทำโครงงานนั้น โปรแกรม Arduino IDE นั้นมีความเข้าใจยากสำหรับผู้ดจัดทำที่มีพื้นฐานในการเขียนคำสั่งน้อย รวมไปถึงระยะเวลาที่จะศึกษาโปรแกรให้เข้าใจอย่างลึกซึ้ง ดังนั้นต้องใช้เวลามากพอสมควรเพื่อให้ผู้จัดทำสิ่งประดิษฐ์นี้ให้มีความสมบูรณ์พร้อมทุกอย่าง

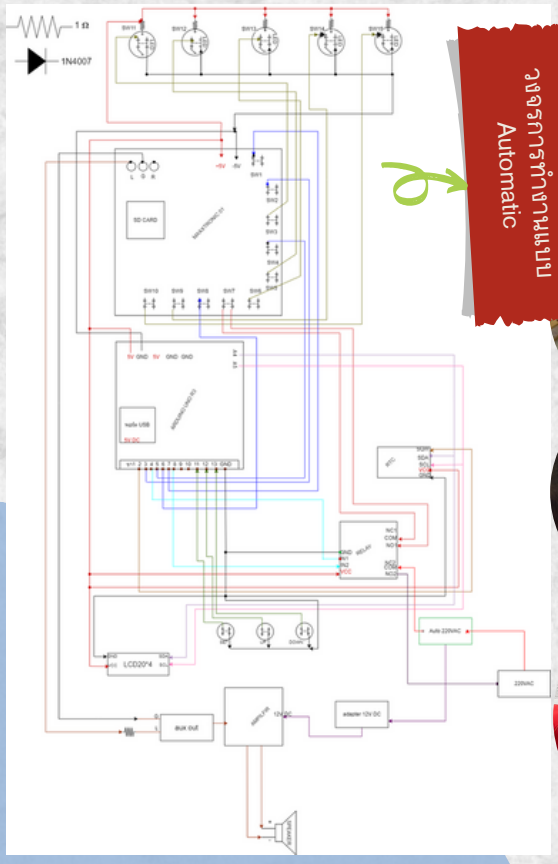
หลักการทำงานของเครื่องประกาศคำสั่ง



หลักการทำงานแบบ Automatic

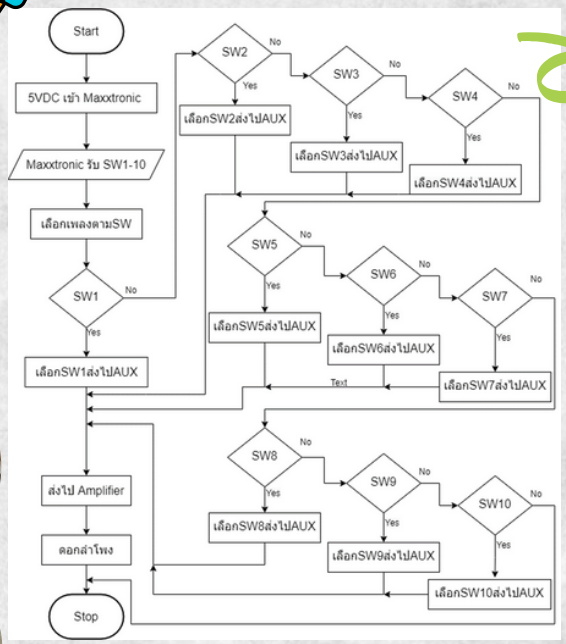


วงจรการทำงานของเครื่องประกาศคำสั่งอัตโนมัติ

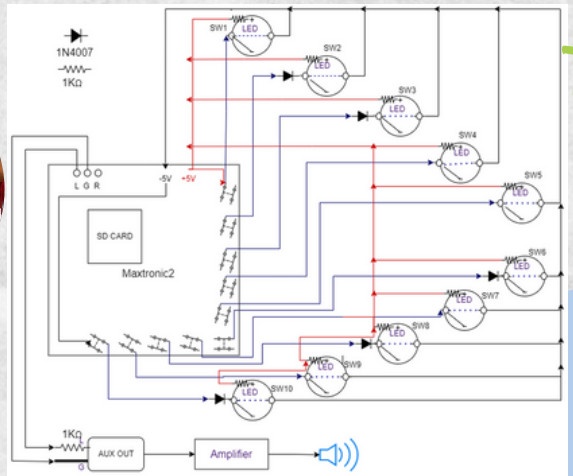


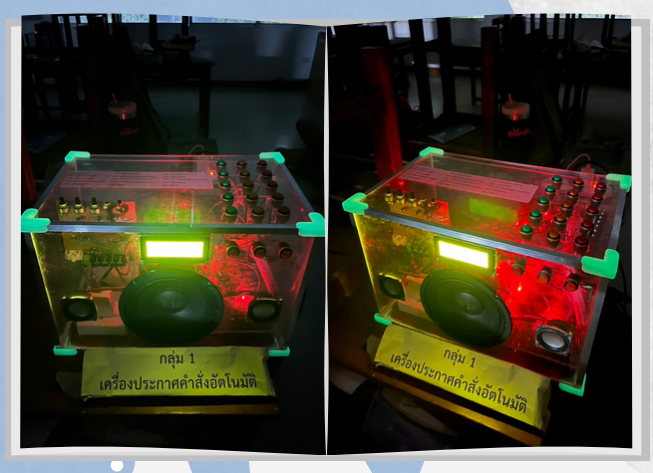
หลักการทำงานแบบ Automatic

หลักการทำงานแบบ Manual



หลักการทำงานแบบ Manual





สมาชิกโครงงาน

นรจ.นราวิทย์ ชูราษฎ์	(หน.กลุ่ม)
นรจ.สิทธิโชค แก้วสระแสน	(รองหน.กลุ่ม)
นรจ.ศุภพฏฐ์ บุญซ้อน	(สมาชิกกลุ่ม)
นรจ.วรภัทร สิงหะ	(สมาชิกกลุ่ม)
นรจ.ปฏิวัติ นอกไธสง	(สมาชิกกลุ่ม)

ครูที่ปรึกษา

น.ต.ฐากร สร้อยมณี

จ.อ.บุญญฤทธิ์ บุญวัง