

ระบบแจ้งเตือนการเสื่อมสภาพกรองอากาศของเครื่องกรองอากาศ

AIR FILTER DETERIORATION NOTIFICATION SYSTEM FOR AIR PURIFIER



พสุธิดา วิสิทธิกรณ์
PASUTHIDA WISITTHIKORN

ศุภกานต์ ภาริบุตร
SUPAKAN PARIBUT

ปริญญาานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
ภาควิชาวิศวกรรมอิเล็กทรอนิกส์
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
พ.ศ.2566

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบแจ้งเตือนการเสื่อมสภาพกรองอากาศของเครื่องกรองอากาศ

AIR FILTER DETERIORATION NOTIFICATION SYSTEM FOR AIR PURIFIER



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ.2566

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาานิพนธ์ปีการศึกษา 2566

ภาควิชา วิศวกรรมอิเล็กทรอนิกส์

คณะ วิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ระบบแจ้งเตือนการเสื่อมสภาพกรองอากาศของเครื่องกรองอากาศ

AIR FILTER DETERIORATION NOTIFICATION SYSTEM FOR AIR PURIFIER

ผู้จัดทำ นางสาวพสุธิตา วิสิทธิ์กรณ์ รหัสประจำตัว 63010661

นางสาวศุภกานต์ ฝาริบุตร รหัสประจำตัว 63010928

ปริญญาานิพนธ์นี้ผ่านการตรวจสอบโดยอาจารย์ที่ปรึกษาแล้ว



(รองศาสตราจารย์ จิรวัดน์ ปานกลาง)

อาจารย์ที่ปรึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อโครงการ	ระบบแจ้งเตือนการเสื่อมสภาพกรองอากาศของเครื่องกรองอากาศ	
นักศึกษา	นางสาวสุชิตา วิสิทธิ์กรณ	รหัสนักศึกษา 63010661
	นางสาวศุภกานต์ ฝาริบุตร	รหัสนักศึกษา 63010928
ปริญญา	วิศวกรรมศาสตรบัณฑิต	
ภาควิชา	วิศวกรรมอิเล็กทรอนิกส์	
ปีการศึกษา	2566	
อาจารย์ที่ปรึกษาโครงการ	รองศาสตราจารย์ จิรวัดน์ ปานกลาง	

บทคัดย่อ

รายงานฉบับนี้เป็นส่วนหนึ่งของรายวิชา Project 2 จะกล่าวถึงระบบการแจ้งเตือนการเสื่อมสภาพกรองอากาศของเครื่องกรองอากาศ โดยจะบันทึกค่าความหนาแน่นของฝุ่นขาเข้าและค่าความหนาแน่นของฝุ่นขาออกที่ Optical Dust Sensor PM2.5 (GP2Y1014AU0F) สามารถตรวจจับได้ และบันทึกค่าแรงดันไฟฟ้าที่ DC Fan วัดค่าได้เข้าไปเก็บไว้ที่ Google Sheets รวมไปถึงมีการแจ้งเตือนข้อมูลผ่านแอปพลิเคชัน Line ซึ่งการที่จะเก็บข้อมูลไว้ที่ Google Sheet และมีการแจ้งเตือนทางแอปพลิเคชัน Line ผ่าน Line Notify ได้นั้นจะต้องมีการเขียนโปรแกรมผ่าน Arduino IDE และ Apps Script

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Project Title	Air Filter Deterioration Notification System For Air Purifier	
Student	Miss Pasuthida Wisitthikorn	Student ID 63010661
	Miss Supakan Paribut	Student ID 63010928
Degree	Bachelor of Engineering	
Program	Electronics Engineering	
Year	2023	
Project Advisor	Assoc. Prof. Jirawat Panklang	

ABSTRACT

This report is part of Project 2 and discusses the Air filter deterioration notification system for air purifier. It records the inlet dust density and outlet dust density detected by the Optical Dust Sensor PM2.5 (GP2Y1014AU0F), as well as the voltage that measure by the DC fan. These values are stored in Google Sheets, and notifications are sent via the Line application using Line Notify. To accomplish this, programming is done using Arduino IDE and Apps Script.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

โครงการเรื่องระบบแจ้งเตือนการเสื่อมสภาพของเครื่องกรองอากาศสำเร็จไปได้ด้วยดี โดยได้รับการช่วยเหลือ ให้ความรู้ แนะนำวิธีการแก้ไขปัญหามาจาก รศ.จิรวัดน์ ปานกลาง อาจารย์ที่ปรึกษา โครงการ และเพื่อน ๆ ภาควิชาวิศวกรรมอิเล็กทรอนิกส์ชั้นปี 4 ที่คอยช่วยแนะนำข้อมูลสำหรับโครงการ และได้มอบกำลังใจที่ดีตลอดการทำโครงการนี้ นอกจากนี้ผู้จัดทำขอขอบพระคุณภาควิชาวิศวกรรม อิเล็กทรอนิกส์ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าจตุรทิศมาด้วย

ผู้จัดทำหวังเป็นอย่างยิ่งว่าโครงการเรื่องนี้เป็นประโยชน์แก่ผู้ที่สนใจศึกษาเกี่ยวกับระบบแจ้งเตือน การเสื่อมสภาพของเครื่องกรองอากาศ หากผิดพลาดประการใด คณะผู้จัดทำขออภัยมา ณ ที่นี้ด้วย



พสุธิตา วิสิทธิ์กรณ

ศุภกานต์ ฝาริบุตร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

เรื่อง	หน้า
บทคัดย่อ.....	I
บทคัดย่อ (ภาษาอังกฤษ).....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
สารบัญตาราง.....	VI
สารบัญรูป.....	VII
บทที่ 1 บทนำ	1
1.1 ที่มาและความสำคัญของโครงการ.....	1
1.2 วัตถุประสงค์ของโครงการ	1
1.3 ขอบเขตในการจัดทำโครงการ.....	1
1.4 ขอบเขตการทำงาน.....	1
1.5 ประโยชน์ที่คาดว่าจะได้รับ.....	2
1.6 ระยะเวลาในการดำเนินงาน	2
บทที่ 2 ทฤษฎีและเอกสารที่เกี่ยวข้อง	3
2.1 เครื่องกรองอากาศ (Air Purifier).....	3
2.1.1 หลักการทำงาน	3
2.1.2 ส่วนประกอบ.....	4
1) มอเตอร์ (Motor).....	4
1.1) มอเตอร์ไฟฟ้ากระแสตรง (Direct Current Motor : DC Motor).....	4
1.2) มอเตอร์ไฟฟ้ากระแสสลับ (Alternating Current Motor : AC Motor).....	5
2) พัดลมแบบหมุนเหวี่ยง (Centrifugal Fan).....	6
3) สายไฟ.....	8
4) แผ่นกรองอากาศ	8
4.1) แผ่นกรองชั้นนอก (Pre-Filters).....	8
4.2) แผ่นกรองอากาศ (HEPA Filter).....	8
4.3) แผ่นกรองคาร์บอน (Carbon Filter)	10
2.2 พัดลมกระแสตรง (Direct Current Fan : DC Fan).....	10
2.2.1 หลักการทำงาน	11

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

เรื่อง	หน้า
2.3 Optical Dust Sensor PM2.5 (GP2Y1014AU0F).....	11
2.3.1 ข้อมูลจำเพาะ.....	11
2.3.2 ขาต่อใช้งาน.....	12
2.3.3 หลักการทำงาน	12
2.3.4 ส่วนประกอบ.....	13
2.4 Wemos D1 Mini – ESP8266.....	14
2.4.1 ข้อมูลทางเทคนิค.....	14
2.4.2 การใช้งานบอร์ด WeMos D1 ESP8266 และตัวอย่างการทำให้เป็น Webserver ผ่าน WiFi.....	15
2.5 NodeMCU ESP8266 (NodeMCU V3)	15
2.5.1 คุณสมบัติ	15
2.6 Google Sheet.....	17
2.6.1 ความสามารถของ Google Sheet	17
2.7 Apps Script.....	17
2.7.1 วิธีการสร้าง Script.....	18
2.8 LINE Notification.....	18
บทที่ 3 หลักการทำงานและการออกแบบ	19
3.1 แผนผังการทำงาน	19
3.2 แผนผังการทำงาน	19
3.2.1 การทำงานของระบบ.....	19
3.2.2 การทำงานของ Wemos D1 Mini ESP8266 ร่วมกับ Optical Dust Sensor.....	20
3.3.2 การทำงานของ NodeMCU ESP8266 (NodeMCU V3) ร่วมกับพัดลมกระแสตรง	21
บทที่ 4 ผลการทดลอง	23
4.1 ทหาความสัมพันธ์ระหว่าง speed เครื่องกรองอากาศ และ Vout.....	23
4.2 การเขียนโปรแกรมและผลลัพธ์การแสดงผล	25
บทที่ 5 สรุปผลการทดลอง	27
5.1 สรุปผลการทดลอง.....	27
5.2 วิจัยารณ์ผลการทดลอง.....	27
5.3 ข้อเสนอแนะ.....	27
เอกสารอ้างอิง	28
ภาคผนวก.....	30

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่	หน้า
1.1 ระยะเวลาในการทำโครงการตั้งแต่ 1 ธันวาคม 2566 จนถึง 5 มีนาคม 2567.....	3
2.1 ข้อมูลจำเพาะของ Optical Dust Sensor PM2.5 (GP2Y1014AU0F)	13
2.2 การเชื่อมต่อของขาต่อใช้งาน	14
2.3 ข้อมูลทางเทคนิค	17
2.4 ข้อมูลทางเทคนิค.....	19
4.1 วัดแรงดัน DC Fan ที่ติดตั้งภายในเครื่องกรองอากาศ (ไม่ใส่แผ่นกรอง)	23
4.2 วัดแรงดัน DC Fan ที่ติดตั้งภายในเครื่องกรองอากาศ (ใส่แผ่นกรอง : HEPA + Carbon)	23
4.3 เพอร์เซ็นต์แรงดันที่ลดลง.....	24

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่	หน้า
2.1 เครื่องกรองอากาศ.....	3
2.2 รูปแบบวงจรของมอเตอร์ไฟฟ้าแบบอนุกรม	4
2.3 รูปแบบวงจรของมอเตอร์ไฟฟ้าแบบขนาน	4
2.4 รูปแบบวงจรของมอเตอร์ไฟฟ้าแบบผสม	5
2.5 ซิงโครนัสมอเตอร์	5
2.6 อะซิงโครนัสมอเตอร์	6
2.7 พัดลมแบบหมุนเหวี่ยง	6
2.8 Forward Curve.....	6
2.9 Backward Curve.....	7
2.10 Backward Inclined.....	7
2.11 Radial Tipped.....	7
2.12 Radial.....	7
2.13 Pre - Filter.....	8
2.14 HEPA Filter.....	9
2.15 การทำงานของ HEPA.....	10
2.16 Carbon Filter.....	10
2.17 พัดลมกระแสตรง	10
2.18 Optical Dust Sensor PM 2.5 (GP2Y1014AU0F).....	11
2.19 ขาต่อใช้งานของ Optical Dust Sensor PM2.5 (GP2Y1014AU0F)	12
2.20 การทำงานของ Optical Dust Sensor PM2.5 (GP2Y1014AU0F).....	12

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่	หน้า
2.21 ส่วนประกอบส่วนที่ 1 ของ Optical Dust Sensor PM2.5 (GP2Y1014AU0F).....	13
2.22 ส่วนประกอบส่วนที่ 2 ของ Optical Dust Sensor PM2.5 (GP2Y1014AU0F).....	13
2.23 Wemos D1 Mini - ESP8266	14
2.24 NodeMCU ESP8266 (NodeMCU V3)	15
2.25 รูปบอร์ด NodeMCU V3 และขาต่างๆ.....	16
2.26 Google Sheet.....	17
2.27 App Script.....	17
2.28 Line Notify.....	18
3.1 แผนภาพ Flowchart การทำงานของระบบ.....	19
3.2 แผนภาพ Flowchart การทำงานของ Wemos D1 Mini ร่วมกับ Optical Dust Sensor.....	20
3.3 แผนภาพ Flowchart การทำงานของ NodeMCU ESP8266 ร่วมกับพัดลมกระแสตรง.....	21
3.4 Schematic ของ ระบบ	22
4.1 กราฟแสดงความสัมพันธ์ระหว่าง speed เครื่องกรองอากาศ และ V_{out} (ไม่ใช่แผ่นกรอง)	23
4.2 กราฟแสดงความสัมพันธ์ระหว่าง speed เครื่องกรองอากาศ และ V_{out} (ใส่แผ่นกรอง)	24
4.3 กราฟการเปรียบเทียบ V_{out} ขณะไม่ใช่แผ่นกรองและใส่แผ่นกรอง	24
4.4 ตัวอย่าง เขียนโปรแกรม Arduino IDE และ Google Sheet รวมไปถึงการแจ้งเตือน LINE Notify.....	25
4.5 ตัวอย่างเงื่อนไขให้แจ้งเตือนที่ LINE Application ผ่าน LINE Notify.....	25
4.6 แสดงข้อมูลที่ได้ที่ Google Sheet.....	26
4.7 ข้อมูลที่แจ้งเตือนจากเงื่อนไขที่มีการแจ้งเตือนผ่าน LINE Notify เมื่อค่าความหนาแน่นน้อยกว่า 420.....	26

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

1.1 ที่มาและความสำคัญของโครงการ

ปัจจุบันปัญหาฝุ่นมีมากเพิ่มขึ้น ไม่ว่าจะเป็นฝุ่น PM2.5 ฝุ่นจากมลพิษต่าง ๆ หรือฝุ่นที่มนุษย์สร้างขึ้น ดังนั้น ทุกบ้านหรือทุกสถานที่ส่วนใหญ่ล้วนต้องมีเครื่องกรองอากาศ เพื่อลดปัญหาสูดอากาศที่มีฝุ่นขนาดเล็ก หรือเพื่อต้องการสูดอากาศสะอาดลดปัญหาสุขภาพ โดยเครื่องกรองอากาศส่วนใหญ่จะประกอบด้วย กรองอากาศ พัดลม ตัวควบคุม ซึ่งวิธีปรับความเร็วพัดลมนั้นจะมีอิทธิพลต่อประสิทธิภาพของเครื่องฟอกอากาศภายในบ้าน และการระบายอากาศที่ดีนั้นไม่ใช่ว่าการที่ยิ่งมีความเร็วพัดลมมากแล้วประสิทธิภาพการกรองอากาศจะสูง

ปัญหาสำคัญคือ การที่เราใช้เครื่องกรองอากาศตลอดเวลา ทำให้ประสิทธิภาพของเครื่องกรองอากาศลดลง เนื่องจาก ฝุ่นในปัจจุบันมีปริมาณเพิ่มมากขึ้น ในขณะที่ใช้เครื่องกรองอากาศนั้น เราไม่สามารถที่จะรู้ได้เลยว่าแผ่นกรองที่อยู่ในเครื่องตันหรือไม่ ด้วยเหตุนี้ หากแผ่นกรองตันแล้วยังใช้งานต่อเนื่อง สามารถทำให้เครื่องกรองอากาศพังหรือเสียหายได้

โครงการนี้จัดทำขึ้นโดย ตรวจสอบค่าฝุ่นและนำค่าฝุ่นที่วัดได้มาใช้อ้างอิง กล่าวคือเมื่อมีฝุ่นมาก พัดลมในเครื่องกรองอากาศต้องใช้ความเร็วสูงสุดเพื่อที่จะสามารถกรองอากาศหรือกรองฝุ่นได้อย่างมีประสิทธิภาพ อีกทั้งออกแบบโปรแกรมจัดเก็บข้อมูลค่าฝุ่นขาเข้า ค่าฝุ่นขาออก แจ้งเตือนผู้ใช้ โดยข้อมูลดังกล่าว สามารถบอกถึงประสิทธิภาพของเครื่องกรองอากาศได้ว่า แผ่นกรองยังใช้งานได้ปกติหรือไม่ หรือแผ่นกรองตันจำเป็นต้องเปลี่ยนแผ่นกรอง

1.2 วัตถุประสงค์ของโครงการ

- 1.2.1 เพื่อศึกษาการคิดวิเคราะห์ แก้ไขปัญหา เกี่ยวกับวงจรอิเล็กทรอนิกส์
- 1.2.2 เพื่อออกแบบเครื่องกรองอากาศให้สามารถตรวจสอบค่าฝุ่นขาเข้า ฝุ่นขาออก
- 1.2.3 เพื่อออกแบบเครื่องกรองอากาศให้สามารถแจ้งเตือนประสิทธิภาพของแผ่นกรองอากาศได้

1.3 สมมุติฐานของการศึกษา

1.3.1 สามารถออกแบบเครื่องกรองอากาศให้เก็บบันทึกข้อมูลฝุ่นขาเข้า ฝุ่นขาออก แรงดันของ DC Fan จัดเก็บที่ Google Sheets

1.3.2 สามารถออกแบบให้แจ้งเตือนค่าฝุ่น ผ่าน Line Notify โดยใช้เทคโนโลยี Wireless Sensor Network

1.4 ขอบเขตของโครงการ

- 1.4.1 ออกแบบวงจรโดยใช้ WeMos D1 mini เป็นไมโครคอนโทรลเลอร์
- 1.4.2 ควบคุมการทำงาน และการสื่อสารระหว่างอุปกรณ์อิเล็กทรอนิกส์ด้วยเทคโนโลยี Wireless Sensor Network เพื่อนำมาประยุกต์ใช้เพื่อเก็บรวบรวมข้อมูลการใช้ไฟฟ้าของเครื่องใช้ไฟฟ้า
- 1.4.3 กำหนดขอบเขตการทำงานของระบบให้ทำงานตามวัตถุประสงค์
- 1.4.4 เขียนโปรแกรมรายงานผ่าน Network นำค่าจาก Sensor เก็บข้อมูล
- 1.4.5 ออกแบบโปรแกรมแจ้งเตือนข้อมูล ให้แจ้งเตือนไปยัง Application

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.5 ประโยชน์ที่คาดว่าจะได้รับ

- 1.5.1 เข้าใจหลักการทำงานของเครื่องฟอกอากาศ
- 1.5.2 ได้ฝึกฝนทักษะในการปฏิบัติงานด้านอิเล็กทรอนิกส์
- 1.5.3 สามารถวิเคราะห์หรือแก้ไขปัญหาที่เกิดขึ้นได้
- 1.5.4 ฝึกฝนการเขียนโปรแกรม เพื่อนำค่ามาแสดงผลใน application
- 1.5.5 ทำให้มีทักษะในการออกแบบ Program เพื่อให้สามารถแสดงผลตามฟังก์ชันที่ต้องการได้

1.6 ระยะเวลาในการดำเนินการ

เริ่มตั้งแต่วันที่ 1 ธันวาคม พ.ศ. 2566 จนถึงวันที่ 5 มีนาคม พ.ศ. 2567

รายละเอียด / สัปดาห์ที่	ธันวาคม				มกราคม					กุมภาพันธ์				มีนาคม			
	1	2	3	4	1	2	3	4	5	1	2	3	4	1	2	3	4
1. ได้รับหัวข้อ โครงการ	←→																
2. ศึกษาหาข้อมูล ที่และ ออกแบบ โครงสร้าง				←→				→									
3. จัดซื้ออุปกรณ์						←→											
4. ทดสอบ โปรแกรม									←→								
5. ทดสอบและ ประกอบวงจร											←→						
6. จัดทำรูปเล่ม รายงาน	←→																

ตารางที่ 1.1 ระยะเวลาในการทำโครงการตั้งแต่ 1 ธันวาคม 2566 จนถึง 5 มีนาคม 2567

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2 ทฤษฎีและเอกสารที่เกี่ยวข้อง

2.1 เครื่องกรองอากาศ



รูปที่ 2.1 เครื่องฟอกอากาศ

เครื่องกรองอากาศ (Air Purifier) เป็นเครื่องใช้ไฟฟ้าที่ออกแบบมาเพื่อช่วยในการกรองสิ่งสกปรก ฝุ่นละออง ไรฝุ่น เชื้อโรค เชื้อรา สารกระตุ้นให้เกิดภูมิแพ้ รวมถึงกลิ่นอันไม่พึงประสงค์ เช่น กลิ่นควันบุหรี่ กลิ่นอับ กลิ่นเหม็น โดยเครื่องกรองอากาศจะดูดอากาศเข้าเครื่องผ่านตัวกรองเพื่อดักจับสิ่งแปลกปลอม ในอากาศเอาไว้ และปล่อยอากาศบริสุทธิ์ออกมาแทน

2.1.1 หลักการทำงาน

หลักการทำงานของเครื่องกรองอากาศจะใช้หลักการดูดอากาศเข้าไปแล้วดักจับสิ่งแปลกปลอม ด้วยแผ่นกรอง หรือฆ่าเชื้อ หรือการกำจัดกลิ่นด้วยระบบอื่น ๆ จากนั้นก็จะปล่อยอากาศที่บริสุทธิ์มากขึ้น ออกมา ซึ่งจะมีส่วนประกอบในการทำงาน 3 ส่วน ดังนี้

1) แผ่นกรอง (Air Filter) มีหน้าที่กรองสิ่งสกปรกต่าง ๆ เช่น ฝุ่นละออง ไรฝุ่น เชื้อโรค เชื้อรา สารกระตุ้นให้เกิดภูมิแพ้ เป็นต้น ซึ่งแผ่นกรองนี้อาจเป็นได้ทั้งเส้นใย กระดาษ หรือตาข่ายก็ได้ โดยในปัจจุบัน แผ่นกรองที่นิยมใช้กันมากที่สุดจะเป็น High Efficiency Particulate (HEPA) ที่เป็นแผ่นกรองที่ผลิตจากใยแก้วนำแสง และมีประสิทธิภาพสูงในการดักจับฝุ่น

2) ระบบกรองอากาศ (Electrostatic Precipitator) เป็นระบบที่ทำงานโดยใช้หลักการไฟฟ้าสถิต โดยจะปล่อยประจุลบออกมาเพื่อจับประจุบวกให้อยู่รวมกันเป็นกลุ่มซึ่งเมื่อมีน้ำหนักมากพอ ก็จะทำให้ตกลงพื้นโดยไม่ฟุ้งกระจายให้เป็นอันตรายต่อร่างกาย

3) ระบบกำจัดกลิ่น (Gas-Phase Air Filter) เป็นระบบที่ใช้ถ่านกัมมันต์ (Activated Carbon) ในการดูดซับกลิ่นไม่พึงประสงค์ รวมไปถึงกลิ่นสารเคมีหรือก๊าซพิษอื่น ๆ ด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.2 ส่วนประกอบ

1) มอเตอร์

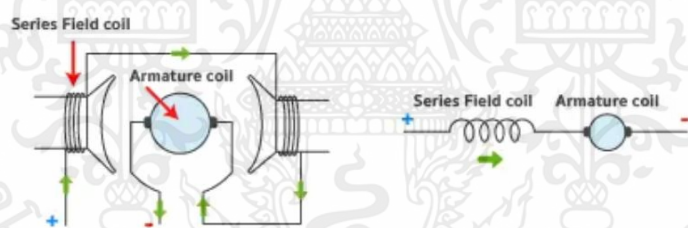
มอเตอร์ไฟฟ้า คือ อุปกรณ์ชนิดหนึ่งที่ทำหน้าที่ในการแปลงพลังงานไฟฟ้าเป็นพลังงานกล โดยเมื่อจ่ายกระแสไฟฟ้าให้มอเตอร์ไฟฟ้าแล้วกระแสไฟฟ้าจะไหลผ่านขดลวดที่อยู่ภายในมอเตอร์ จากนั้นจะเกิดการเหนี่ยวนำให้เกิดสนามแม่เหล็กเพื่อผลักให้แกนของมอเตอร์หมุนได้ โดยมอเตอร์ไฟฟ้าที่นิยมใช้งานในอุตสาหกรรมจะแบ่งตามชนิดแหล่งจ่ายไฟได้ 2 ประเภท ดังนี้

1.1) มอเตอร์ไฟฟ้ากระแสตรง (Direct Current Motor : DC Motor)

มอเตอร์ไฟฟ้ากระแสตรงจะทำงานด้วยไฟฟ้ากระแสตรง (Direct Current) โดยจะมีข้อดีคือสามารถควบคุมความเร็วของมอเตอร์ได้ง่ายและถูกต้องแม่นยำกว่าแบบมอเตอร์ไฟฟ้ากระแสสลับ (AC motor) นอกจากนี้มอเตอร์ไฟฟ้ากระแสตรงยังสามารถจำแนกออกได้อีก 3 ชนิด ดังนี้

1.1.1) มอเตอร์ไฟฟ้าแบบอนุกรม (Series Motor)

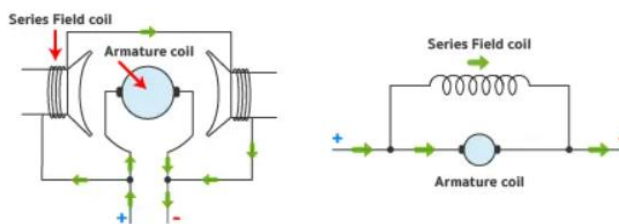
วงจรของมอเตอร์ไฟฟ้าแบบอนุกรมจะต่ออนุกรมกันทั้งขดลวดสนามแม่เหล็ก (Field Coil) และขดลวดอาร์เมเจอร์ (Armature Coil) ซึ่งกระแสไฟฟ้าที่ไหลผ่านขดลวดทั้งสองนั้นจะมีค่าเท่ากัน โดยปริมาณกระแสจะขึ้นอยู่กับภาระโหลดที่แกนมอเตอร์ ซึ่งมอเตอร์แบบอนุกรมจะมีแรงบิดที่สูงมากแต่ความเร็วจะช้าลงเมื่อมีโหลดที่มากขึ้นบริเวณแกนมอเตอร์ นิยมใช้กับงานที่ต้งรับโหลดมาก ๆ เช่น มอเตอร์ยกของ มอเตอร์ขับเคลื่อนรถไฟฟ้า



รูปที่ 2.2 รูปแบบวงจรของมอเตอร์ไฟฟ้าแบบอนุกรม

1.1.2) มอเตอร์ไฟฟ้าแบบขนาน (Shunt Motor)

วงจรของมอเตอร์ไฟฟ้าแบบขนาน คือ มอเตอร์ที่ขดลวดสนามแม่เหล็ก (Field Coil) จะต่อขนานกับขดลวดอาร์เมเจอร์ (Armature Coil) ซึ่งกระแสที่จ่ายให้ขดลวดแต่ละชุดจะไม่เท่ากัน มอเตอร์ไฟฟ้ากระแสตรงแบบขนาน (Shunt Motor) จะให้แรงบิดและความเร็วที่คงที่โดยไม่ขึ้นกับโหลด แต่แรงบิดที่ได้จะไม่สูงเท่ามอเตอร์ไฟฟ้าแบบอนุกรม นิยมนำไปใช้กับงานที่ต้องการแรงบิดและความเร็วคงที่ เช่น มอเตอร์ของเครื่องเจาะ มอเตอร์ของเครื่องกลึง

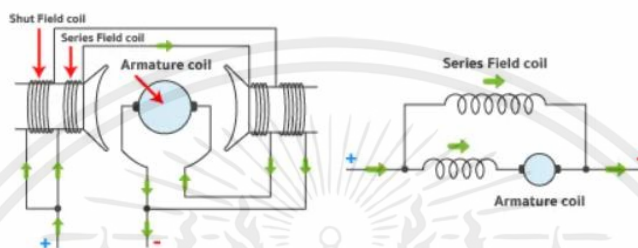


รูปที่ 2.3 รูปแบบวงจรของมอเตอร์ไฟฟ้าแบบขนาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.1.3) มอเตอร์ไฟฟ้าแบบผสม (Compound Motor)

วงจรของมอเตอร์ไฟฟ้าแบบผสมจะมีขดลวดสนามแม่เหล็ก (Field Coil) 2 ชุด โดยจะนำกระแสไฟฟ้าที่ไหลผ่านขดลวดสนามแม่เหล็กแบบอนุกรม (Series Field Coil) และขดลวดอาร์เมเจอร์ (Armature Coil) กลับมาขนานกับขดลวดสนามแม่เหล็กแบบขนาน (Shunt Field Coil) โดยมอเตอร์ไฟฟ้าแบบผสมจะรวมคุณสมบัติเด่นของมอเตอร์ไฟฟ้าแบบอนุกรม (Series motor) และมอเตอร์ไฟฟ้าแบบขนาน (Shunt Motor) เข้าด้วยกัน ทำให้มอเตอร์ไฟฟ้าแบบผสมมีแรงบิดมากกว่ามอเตอร์ไฟฟ้าแบบขนาน (Shunt Motor) แต่ไม่เท่ากับมอเตอร์ไฟฟ้าแบบอนุกรม (Series motor) และมีความคงที่ที่ดีกว่ามอเตอร์ไฟฟ้าแบบอนุกรม (Series motor) แต่ไม่ดีเท่ามอเตอร์ไฟฟ้าแบบขนาน (Shunt Motor)



รูปที่ 2.4 รูปแบบวงจรของมอเตอร์ไฟฟ้าแบบผสม

1.2) มอเตอร์ไฟฟ้ากระแสสลับ (Alternating Current Motor : AC Motor)

มอเตอร์ไฟฟ้ากระแสสลับจะทำงานด้วยไฟฟ้ากระแสสลับ (Alternating Current) มีทั้งที่ทำงานด้วยไฟฟ้ากระแสสลับชนิด 1 เฟส และไฟฟ้ากระแสสลับชนิด 3 เฟส ซึ่งมีข้อดีคือราคาถูกกว่าเมื่อเทียบกับมอเตอร์ไฟฟ้ากระแสตรงที่กำลังเท่ากัน นอกจากนี้ยังมีโครงสร้างที่เรียบง่ายไม่ซับซ้อนและทนทานทำให้แทบไม่ต้องบำรุงรักษา ทั้งนี้มอเตอร์ไฟฟ้ากระแสสลับยังสามารถจำแนกออกได้อีก 2 ชนิด ดังนี้

1.2.1) ซิงโครนัสมอเตอร์ (Synchronous Motor)

ซิงโครนัสมอเตอร์ (Synchronous Motor) นั้นจะมีจุดเด่นที่มีความเร็วคงที่ตลอดเวลาแม้ว่าโหลดจะมีการเปลี่ยนแปลง โดยความเร็วรอบจะขึ้นอยู่กับความถี่ของไฟฟ้ากระแสสลับที่จ่ายเข้าไป ซึ่งตัวแกนโรเตอร์ของซิงโครนัสมอเตอร์จะหมุนด้วยความเร็วซิงโครนัส และส่วนที่เป็นโรเตอร์จะทำหน้าที่สร้างสนามแม่เหล็กเพื่อผลักตัวเองให้หมุนไปพร้อมกับสนามแม่เหล็กที่เกิดขึ้นที่ตัว Stator ซิงโครนัสมอเตอร์จะเหมาะกับระบบที่ต้องการความเร็วที่คงที่ เช่น แขนกลในงานอุตสาหกรรมหรือระบบอัตโนมัติ



รูปที่ 2.5 ซิงโครนัสมอเตอร์

1.2.2) อะซิงโครนัสมอเตอร์ (Asynchronous Motor)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อะซิงโครนัสมอเตอร์ (Asynchronous Motor) หรือเรียกอีกอย่างว่า มอเตอร์เหนี่ยวนำ (Induction Motor) คือ มอเตอร์ไฟฟ้ากระแสสลับที่ใช้พลังงานไฟฟ้าเหนี่ยวนำให้เกิดการสร้างสนามแม่เหล็กที่ตัวแกน Rotor เพื่อปลั๊กตัวเองให้หมุน โดยจะส่งผ่านสนามแม่เหล็กไฟฟ้าจาก Stator มายัง Rotor ที่มีขดลวดพันอยู่ อะซิงโครนัสมอเตอร์จะถูกนำมาใช้อย่างแพร่หลายที่สุดในงานอุตสาหกรรม



รูปที่ 2.6 อะซิงโครนัสมอเตอร์

2) พัดลมแบบหมุนเหวี่ยง (Centrifugal Fan)



รูปที่ 2.7 พัดลมแบบหมุนเหวี่ยง

พัดลมแบบหมุนเหวี่ยง (Centrifugal Fan) จะทำงานด้วยหลักการนำลมเข้าจากทิศทางตามแนวแกนกลางของใบพัดซึ่งจะทำให้อากาศที่เข้ามามีระดับความเร็วสูง จากนั้นจึงเกิดแรงหมุนเหวี่ยงเพื่อทำหน้าที่ดันอากาศออกไปทางด้านข้างของตัวพัดลมตามแนวรัศมี โดยพัดลมแบบหมุนเหวี่ยงนั้น จะแบ่งออกเป็น 5 ประเภท ดังนี้

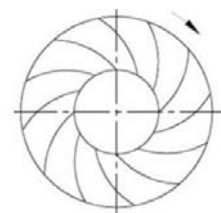


FORWARD CENTRIFUGAL
BLADES FACING FORWARD TO
DIRECTION OF ROTATION

รูปที่ 2.8 Forward Curve

2.1) Forward Curve เป็นพัดลมแบบใบพัดโค้งไปข้างหน้า ในทิศทางเดียวกับการหมุนของชุดใบพัดซึ่งมีลักษณะคล้ายกับกรงกระรอก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



**BACKWARD CENTRIFUGAL
BLADES FACING BACKWARDS TO
DIRECTION OF ROTATION**

รูปที่ 2.9 Backward Curve

2.2) Backward Curve เป็นพัดลมแบบใบพัดโค้งไปข้างหลังในทิศทางตรงข้ามกับการหมุนของชุดใบพัดซึ่งมีลักษณะโค้ง



**Backward Inclined
Blades**

รูปที่ 2.10 Backward Inclined

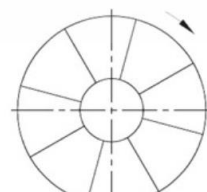
2.3) Backward Inclined เป็นพัดลมแบบใบพัดตรงเอียงไปข้างหลังในทิศทางตรงข้ามกับการหมุนของใบพัด



**Radial Tipped
Blades**

รูปที่ 2.11 Radial Tipped

2.4) Radial Tipped เป็นพัดลมแบบใบพัดเอียงไปข้างหน้าในทิศทางเดียวกับการหมุนของใบพัด



**RADIAL BLADES
BLADES PERPENDICULAR TO AXIS
OF ROTATION**

รูปที่ 2.12 Radial

2.5) Radial เป็นพัดลมแบบใบพัดตรง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) สายไฟ

สายไฟ เป็นส่วนประกอบที่สำคัญอย่างหนึ่งในระบบไฟฟ้า ทำหน้าที่ส่งผ่านพลังงานหรือสัญญาณไฟฟ้าจากจุดหนึ่งไปยังอีกจุดหนึ่งโดยเฉพาะระบบส่งจ่ายกำลังไฟฟ้าจากแหล่งผลิตไฟฟ้าไปยังผู้ใช้งานไฟฟ้าทั่วประเทศผ่านระบบสายส่งและระบบจำหน่ายไฟฟ้า ทั้งในระบบแรงดันสูง แรงดันปานกลาง และแรงดันต่ำ นอกจากนี้สายไฟยังใช้ในระบบสื่อสารและโทรคมนาคม และระบบควบคุมในภาคอุตสาหกรรมอีกด้วย ทั้งนี้สายไฟฟ้า คือ วัสดุที่ประกอบไปด้วยธาตุโลหะที่มีคุณสมบัติในการนำไฟฟ้าและนำความร้อนได้ดี เนื่องจากเนื้อโลหะที่มีความแข็งและเหนียว โดยเฉพาะทองแดงที่สามารถนำมาแปรรูปได้ตามต้องการ จึงได้รับความนิยมในวงการของอุตสาหกรรมซึ่งสายไฟแต่ละชนิดจะได้รับการออกแบบแตกต่างกันออกไปตามโครงสร้างและคุณสมบัติการใช้งาน เช่น สายที่ประกอบไปด้วยตัวนำไฟฟ้าเพียงอย่างเดียว, สายที่ประกอบด้วยฉนวนหุ้มตัวนำไฟฟ้า, สายที่ประกอบด้วยเปลือกหุ้มหรือชั้นป้องกันเสริมเป็นส่วนประกอบอยู่ภายใน

4) แผ่นกรองอากาศ

4.1) แผ่นกรองชั้นนอก (Pre-Filters)

แผ่นกรองชั้นแรกเป็นวิธีที่ประหยัดในการประหยัดเวลาและเงินในการเปลี่ยนแผ่นกรองอากาศในครัวเรือน เนื่องจากเป็นตัวกรองแรกที่อากาศต้องผ่านเข้าไปในเครื่องกรองอากาศ และมีความสำคัญอย่างมากเพราะช่วยป้องกันไม่ให้อนุภาคขนาดใหญ่ เช่น เส้นผม ไปถึงตัวกรองหลัก ซึ่งจะเพิ่มประสิทธิภาพและอายุการใช้งานของตัวกรองอื่น ๆ รวมไปถึงส่วนประกอบภายใน และช่วยให้มั่นใจว่าระบบกรองอากาศอยู่ในสภาพที่เหมาะสมที่สุด โดยปกติแล้วสามารถล้าง ดูดฝุ่น หรือเปลี่ยนใหม่ได้



รูปที่ 2.13 Pre – Filter

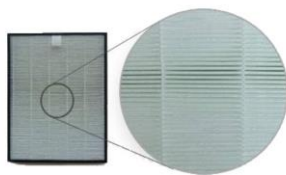
<https://www.dwacool.com/wp-content/uploads/2021/12/Pre-filter-3.jpg>

4.2) แผ่นกรองอากาศ (HEPA Filter)

HEPA ย่อมาจาก High Efficiency Particulate Air โดยตัวกรองเหล่านี้เหมาะอย่างยิ่งสำหรับการดักจับอนุภาคที่มีขนาดเล็กมาก ๆ เช่น ละอองเกสร เชื้อรา หรือฝุ่นละอองที่ลอยอยู่ในอากาศ ดังนั้น HEPA Filter จึงเป็นที่ชื่นชอบของบุคคลผู้ที่เป็นโรคมะเร็งแพ้หรือโรคหอบหืด และตัวกรองเหล่านี้จะทำจากวัสดุกรองหลายชั้นทำให้อนุภาคขนาดใหญ่ไม่สามารถผ่านช่องเปิดของเส้นใยและจะถูกจับไว้ได้ทันทีด้วยวัสดุที่ทำจากเทคโนโลยีเส้นใยชั้นสูง ทำให้มั่นใจได้ว่าฝุ่นที่มีขนาดเล็กมาก ๆ รวมไปถึงสิ่งสกปรกอื่น ๆ ก็จะไม่สามารถผ่านไปได้อีก นอกจากนี้แผ่นกรอง HEPA จะสามารถดักจับฝุ่นละอองอนุภาคขนาด 0.3 ไมครอนได้ถึง 99.97% หรือมากกว่านั้น และยังอากาศไหลผ่านแผ่นกรองมากเท่าไร อากาศก็จะยิ่งสะอาดมากขึ้นเท่านั้น เมื่อแผ่นกรอง HEPA เต็ม อากาศจะไม่สามารถผ่านเข้าไปได้อีกต่อไปและจะต้องเปลี่ยนแผ่นกรองใหม่ อย่างไรก็ตามตัวกรอง HEPA มักมีอายุการใช้งาน 2 ถึง 4 ปี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DWA
COOL



HEPA Filter

รูปที่ 2.14 HEPA Filter

<https://www.dwacool.com/wp-content/uploads/2021/12/HEPA-Filter.jpg>

4.2.1) ชนิดของแผ่นกรองอากาศ HEPA

แผ่นกรองอากาศ HEPA นั้นสามารถแยกเป็นชนิดย่อยได้อีก 8 คลาส โดยทางสหภาพยุโรปได้จำแนกไว้ ดังนี้

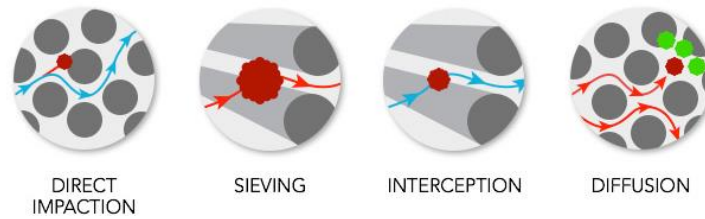
- E10 Class : ดักจับฝุ่นได้ 85%
- E11 Class : ดักจับฝุ่นได้ 95%
- E12 Class : ดักจับฝุ่นได้ 99.5%
- H13 Class : ดักจับฝุ่นได้ 99.95%
- H14 Class : ดักจับฝุ่นได้ 99.995%
- U15 Class : ดักจับฝุ่นได้ 99.9995%
- U16 Class : ดักจับฝุ่นได้ 99.99995%
- U17 Class : ดักจับฝุ่นได้ 99.999995%

4.2.3) การทำงานของแผ่นกรอง HEPA

แผ่นกรองอากาศ HEPA+ มีการดักจับฝุ่นละอองหรืออนุภาคขนาดเล็กที่ปนเปื้อนมากับอากาศ โดยสามารถถูกดักจับได้ 4 รูปแบบ ได้แก่

- รูปแบบที่ 1 : การชน (Direct Impaction) อนุภาคที่สกปรกขนาดใหญ่ เช่น ฝุ่นละออง เกสรดอกไม้ เชื้อราบางประเภท ถูกดักติดอยู่กับเส้นใยของแผ่นกรองโดยตรง
- รูปแบบที่ 2 : การคัดกรอง (Sieving) อนุภาคที่สกปรกขนาดใหญ่ไม่ได้มาชนเส้นใยของแผ่นกรองโดยตรง แต่ติดอยู่ระหว่างช่องว่างระหว่างเส้นใยทำให้ไม่สามารถผ่านแผ่นกรองไปได้
- รูปแบบที่ 3 : การสกัดกั้น (Interception) อนุภาคที่สกปรกขนาดเล็กสามารถเคลื่อนไหวผ่านเส้นใยของแผ่นกรองอากาศด่านแรก ๆ ไปได้ แต่ด้วยแรงเฉื่อยและความหนาแน่นของเส้นใยทำให้อนุภาคขนาดเล็กติดอยู่ด้านข้างของเส้นใย ไม่สามารถผ่านแผ่นกรองไปได้
- รูปแบบที่ 4 : การแพร่ (Diffusion) อนุภาคที่สกปรกขนาดเล็กมาก เช่น ฝุ่นที่มีขนาดเล็กกว่า 2.5 ไมครอน มักมีการเคลื่อนไหวแบบไร้ทิศทาง ทำให้มีโอกาสจะไปชนกับเส้นใยของแผ่นกรองอากาศด้วยตัวเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.15 การทำงานของ HEPA

<https://www.cowaymega.com/what-is-a-hepa-filter/>

4.3) แผ่นกรองคาร์บอน (Carbon Filter)

คาร์บอนนอกจากจะมีประโยชน์ในดินสอแล้วก็ยังเป็นวัสดุกรองที่ดีเยี่ยมเช่นกัน เนื่องจากลักษณะรูพรุนของเม็ดคาร์บอนมีพื้นที่ผิวขนาดใหญ่ที่มีรูพรุนจำนวนมาก ซึ่งเมล็ดพืชหรือคาร์บอนสามารถดูดซับสารปนเปื้อนโดยตรงจากอากาศในระหว่างกระบวนการกรอง ทำให้เหมาะสำหรับการกรองควันหรือกลิ่นที่ไม่พึงประสงค์หรือก๊าซพิษต่าง ๆ ที่ตัวกรอง HEPA ไม่สามารถกรองกลิ่นได้ นอกจากนี้คาร์บอนยังสามารถบำบัดด้วยสารเคมีปฏิกิริยาต่าง ๆ ซึ่งจะทำให้การกรองประเภทนี้มีประสิทธิภาพมากยิ่งขึ้น

DWA
COOL



Carbon Filter

รูปที่ 2.16 carbon Filter

<https://www.dwacool.com/wp-content/uploads/2021/12/Carbon-Filter.jpg>

2.2 พัดลมกระแสตรง (Direct Current Fan : DC Fan)



รูปที่ 2.17 พัดลมกระแสตรง

พัดลมกระแสตรง (Direct Current Fan : DC Fan) เป็นพัดลมที่ทำงานได้ด้วยการส่งกระแสไฟฟ้าจำนวนเล็กน้อยเข้าสู่ขดลวดแม่เหล็กไฟฟ้าแล้วทำให้เกิดสนามแม่เหล็กไฟฟ้าเหนี่ยวนำ จากนั้นจะทำให้เกิดแรงบิดที่แกนของมอเตอร์ส่งผลให้ใบพัดหมุน ซึ่งพัดลมกระแสตรงส่วนใหญ่จะมีแรงดันไฟฟ้าที่ 5VDC, 12VDC, 24VDC และ 48VDC ให้เลือกใช้ และมีขนาดให้เลือกใช้หลายขนาด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.1 หลักการทำงาน

หลักการทำงานของตัวพัดลมด้วยกระแสไฟฟ้าตรง (Direct Current : DC) โดยส่งกระแสไฟฟ้าเข้าสู่ขดลวดแม่เหล็กไฟฟ้าจนเกิดการเหนี่ยวนำแล้วแกนมอเตอร์จะเกิดแรงบิดส่งผลให้ใบพัดลมหมุน อย่างไรก็ตามหากใช้งานกับอุปกรณ์ในประเทศไทยจะต้องมีอะแดปเตอร์หรือหม้อแปลงเพื่อทำการแปลงกระแสไฟฟ้าก่อนแล้วจึงจะทำงานได้อย่างเต็มที่ แต่หากเครื่องจักรหรือระบบควบคุมถูกออกแบบมาเป็นระบบไฟฟ้ากระแสตรงอยู่แล้วก็จะสามารถใช้กับพัดลมกระแสตรงได้เลย

2.3 Optical Dust Sensor PM2.5 (GP2Y1014AU0F)



รูปที่ 2.18 Optical Dust Sensor PM 2.5 (GP2Y1014AU0F)

Optical Dust Sensor PM2.5 (GP2Y1014AU0F) นั้นจะทำหน้าที่เป็นตัวตรวจจับอนุภาคที่มีประสิทธิภาพในการตรวจจับอนุภาคที่มีขนาดเล็กถึง 0.8 ไมครอน โดยจะเห็นได้ว่าสามารถตรวจจับอนุภาคของ PM2.5 ที่มีอนุภาคขนาดเล็กกว่าหรือเท่ากับ 2.5 ไมครอนได้ นอกจากนี้เอาต์พุตของ Optical Dust Sensor PM2.5 เป็นแบบอนาล็อก จึงจำเป็นต้องเชื่อมต่อเข้ากับ ADC ของไมโครคอนโทรลเลอร์ เพื่อวัดค่าอนุภาคในอากาศ

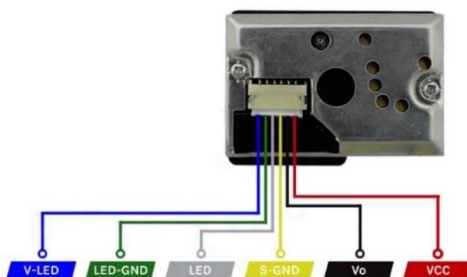
2.3.1 ข้อมูลจำเพาะ

Power supply voltage	5 V to 7 V
Working temperature	-10 °C to 65 °C
Current consumption (maximum)	20 mA
Minimum particle detection value	0.8 microns
Sensitivity	0.5 V / (0.1 mg/m ³)
Clean air voltage (typical)	0.9 V
Storage temperature	-20 °C to 80 °C
Service life	5 years
Size	46 mm x 30 mm x 17.6 mm
Weight	15 g °C

ตารางที่ 2.1 ข้อมูลจำเพาะของ Optical Dust Sensor PM2.5 (GP2Y1014AU0F)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.2 ขาต่อใช้งาน



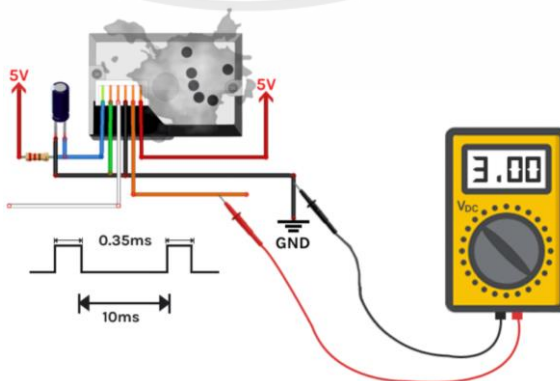
รูปที่ 2.19 ขาต่อใช้งานของ Optical Dust Sensor PM2.5 (GP2Y1014AU0F)

V-LED	เป็นขา V _{CC} ของ LED โดยจะทำการเชื่อมต่อเข้ากับขา 5 V ของ Arduino ด้วยตัวต้านทาน 150 Ω
LED-GND	เป็นขา GND ของ LED โดยจะทำการต่อเข้ากับขา GND ของ Arduino
LED	เป็นขาที่ใช้เพื่อสลับเปิด/ปิด LED โดยจะเชื่อมต่อเข้ากับขาดิจิตอลของ Arduino
S-GND	เป็นขา GND ของ Pulse Dust Sensor Module โดยจะทำการเชื่อมต่อกับขา GND ของ Arduino
V _O	เป็นขาเอาต์พุตของ Dust Sensor Module โดยจะทำการเชื่อมต่อกับขา Analog ของ Arduino
V _{CC}	เป็นขา Power ของ Dust Sensor Module โดยจะทำการเชื่อมต่อเข้ากับ 5 V ของ Arduino

ตารางที่ 2.2 การเชื่อมต่อของขาต่อใช้งาน

2.3.3 หลักการทำงาน

ภายในของ Optical Dust Sensor PM2.5 (GP2Y1014AU0F) ประกอบไปด้วย 3 ส่วนหลัก ได้แก่ ไดโอดเปล่งแสง (แหล่งกำเนิดแสง) โฟโตไดโอด (เครื่องตรวจจับ) และเลนส์คู่หนึ่ง โดย LED และโฟโตไดโอดภายในเซนเซอร์จะถูกวางไว้ในลักษณะที่แกนแสงทั้งสองแกนตัดผ่านพื้นที่การตรวจจับในเซนเซอร์ เมื่อฝุ่นหรือควันเข้าสู่พื้นที่การตรวจจับของเซนเซอร์จะทำให้แสงภายในเซนเซอร์สะท้อนกับฝุ่นหรือควัน เป็นผลให้กระแสไฟฟ้าที่ผลิตโดยโฟโตไดโอดนั้นแตกต่างกันไปตามปริมาณแสงที่ตรวจพบ โดยการแปลงและขยายค่ากระแสไฟฟ้าเป็นค่าแรงดันไฟฟ้าด้วยวงจรที่เหมาะสมจะทำให้ได้เอาต์พุตที่ต้องการจากเซนเซอร์



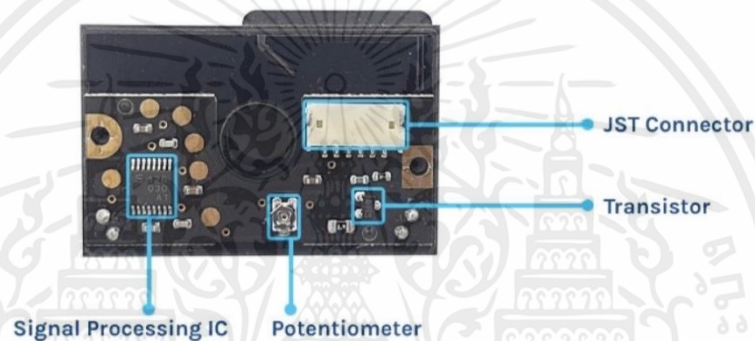
รูปที่ 2.20 การทำงานของ Optical Dust Sensor PM2.5 (GP2Y1014AU0F)

เอกสารนี้เป็นเอกสารทบทวนเนื้อหาสำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การใช้งาน Optical Dust Sensor PM2.5 (GP2Y1014AU0F) จะต้องใช้ขาต่อใช้งาน 2 ขาจาก 6 ขา โดยขาหนึ่งจะกระตุ้น LED และอีกขาหนึ่งจะส่งสัญญาณอนาล็อกออกจากเซนเซอร์ นอกจากนี้ $R_1=150\Omega$ และ $C_1=220\mu F$ เป็นสิ่งจำเป็นสำหรับการไต่รฟ์พัลส์ของ LED หากไม่มีตัวต้านทานและตัวเก็บประจุจะทำให้โมดูลไม่ทำงาน เพื่อให้ได้ข้อมูลที่ต้องการออกจากเซนเซอร์จะต้องทำการกระตุ้น LED ด้วยพัลส์ 10 ms ซึ่งความกว้างของพัลส์ไม่ควรเกิน 0.32 ms และตามคำแนะนำของ datasheet ควรจะสุ่มตัวอย่างสัญญาณหลังจาก 0.28 ms หลังจากกระตุ้นพัลส์ให้กับ LED

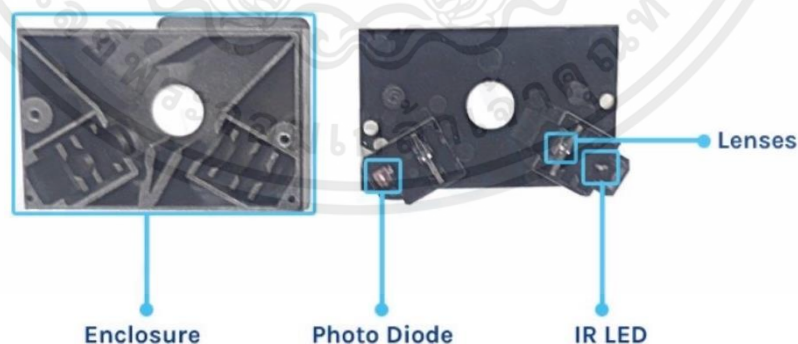
2.3.4 ส่วนประกอบ

โครงสร้างของ Optical Dust Sensor PM2.5 (GP2Y1014AU0F) ยากต่อการบอกส่วนประกอบหลักทั้งหมดของเซนเซอร์ภายในภาพเดียว จึงได้ทำการแบ่งส่วนประกอบของ Optical Dust Sensor PM2.5 (GP2Y1014AU0F) ออกเป็น 2 ส่วน ดังนี้



รูปที่ 2.21 ส่วนประกอบส่วนที่ 1 ของ Optical Dust Sensor PM2.5 (GP2Y1014AU0F)

ในภาพด้านบนจะสามารถเห็น PCB ภายในโมดูลเซนเซอร์เมื่อเปิดฝาครอบด้านบน ซึ่ง PCB นั้นจะมีวงจรรวมประมวลผลหลักเพื่อประมวลผลข้อมูลออกจากโฟโตไดโอด, โฟเทนซิโอมิเตอร์ และทรานซิสเตอร์ที่ไต่รฟ์ LED นอกจากนี้ยังมี JST Connector สำหรับ V_{CC} , GND และข้อมูลเอาต์พุต



รูปที่ 2.22 ส่วนประกอบส่วนที่ 2 ของ Optical Dust Sensor PM2.5 (GP2Y1014AU0F)

ภาพด้านบนจะแสดงให้เห็นว่าเซนเซอร์มีลักษณะอย่างไรเมื่อถอดหรือแกะออก โดยภาพทางด้านซ้ายมือจะเป็นกล่องที่หุ้มโมดูลเซนเซอร์ ส่วนภาพทางด้านขวามือจะเป็นด้านหลังของ PCB ซึ่งบน PCB จะเห็น LED และโฟโตไดโอดติดอยู่ และด้านหน้าของทั้ง LED และโฟโตไดโอดจะมีเลนส์อยู่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 Wemos D1 Mini - ESP8266



รูปที่ 2.23 Wemos D1 Mini - ESP8266

WeMos D1 mini เป็นบอร์ด WiFi ขนาดเล็กที่ใช้ ESP8266EX เป็นตัวประมวลผลหลัก บอร์ดนี้สามารถใช้งานกับ Arduino IDE ได้ จึงสามารถเขียนโปรแกรมด้วย Arduino หรือด้วย Lua compiler ของตัวเองได้ นอกจากนี้ยังสนับสนุนการโปรแกรมผ่านทาง การเชื่อมต่อ Serial และ OTA ด้วย

WeMos D1 mini เป็นบอร์ดพัฒนาขนาดเล็กมีหน่วยความจำ 4MB Flash memory โดยใช้เซตของ ESP8266EX มี digital input ทั้งหมด 11 ขา ทุกขาสามารถใช้งาน Interrupt / PWM / I2C / 1-wire ได้ ยกเว้นขา D0 , WeMos D1 mini มีขา Analog input (ไม่เกิน 3.3V) และมีพอร์ต Micro USB เพื่อเชื่อมต่อกับ PC สำหรับการสื่อสารและโปรแกรมการทำงาน

2.4.1 ข้อมูลทางเทคนิค

Pin	ฟังก์ชัน	ESP-8266 Pin
TX	TXD	TXD
RX	RXD	RXD
A0	Analog input, max 3.3V input	A0
D0	IO	GPIO16
D1	IO,SCL	GPIO5
D2	IO,SDA	GPIO4
D3	IO, 10k Pull up	GPIO0
D4	IO, 10k Pull up, BUILTIN_LED	GPIO2
D5	IO, SCK	GPIO14
D6	IO, MISO	GPIO12
D7	IO, MOSI	GPIO13
D8	IO, 10k Pull-down, SS	GPIO15
G	Ground	GND
5V	5V	-
3V3	3.3V	3.3V
RST	Reset	RST

ตารางที่ 2.3 ข้อมูลทางเทคนิค

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.2 การใช้งานบอร์ด WeMos D1 ESP8266 และตัวอย่างการทำเป็น Webserver ผ่าน WiFi

1. ติดตั้ง USB Driver.
2. ติดตั้ง Arduino IDE.
3. เพิ่ม Additional Board Manager URLs ที่รองรับ ESP8266.
4. เพิ่ม board esp8266 ใน board manager.
5. ทดสอบ Example Blink (ไฟกระพริบ)
6. ทดสอบ Example WiFiWebserver เชื่อมต่อ WiFi เพื่อทำเป็น Webservice.

2.5 NodeMCU ESP8266 (NodeMCU V3)



รูปที่ 2.24 NodeMCU ESP8266 (NodeMCU V3)

NodeMCU ESP8266 เป็นบอร์ดพัฒนาที่มีขนาดเล็กและมีความสามารถในการเชื่อมต่อกับอินเทอร์เน็ตผ่าน WiFi ทำให้เหมาะสำหรับโปรเจกต์ IoT (Internet of Things) ต่างๆ มีความสามารถในการทำงานร่วมกับ Arduino IDE และ PlatformIO เพื่อการพัฒนาโปรแกรมควบคุมอุปกรณ์ต่างๆ ผ่านการเขียนโค้ดด้วยภาษา C/C++ ในบอร์ด NodeMCU ESP8266 มีชิพ ESP8266EX ซึ่งเป็นชิพ WiFi SoC (System-on-Chip) ที่มีประสิทธิภาพสูงและมีราคาที่ถูก มีความสามารถในการทำงานในโหมด Station, Access Point หรือทั้งสองโหมดพร้อมกัน นอกจากนี้ยังมี GPIO (General Purpose Input/Output) ที่สามารถใช้งานได้หลากหลาย สามารถควบคุมอุปกรณ์ภายนอก เช่น หลอด LED, รีเลย์, หรือเซ็นเซอร์ต่างๆ ได้โดยตรง มีการสนับสนุนการเขียนโปรแกรมด้วย Lua และ MicroPython ทำให้สามารถเขียนโปรแกรมได้ด้วยภาษาที่สะดวกและเร็วขึ้น

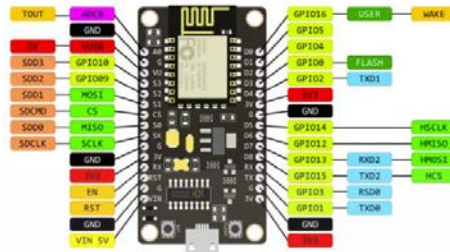
NodeMCU ESP8266 มีขนาดเล็กเหมาะสำหรับใช้ในโปรเจกต์ที่มีพื้นที่จำกัด และมีราคาที่เป็นมิตร ใช้งานง่าย เหมาะสำหรับนักพัฒนาที่ต้องการทำโปรเจกต์ IoT หรือโปรเจกต์ที่ต้องการการเชื่อมต่อกับอินเทอร์เน็ตอย่างง่ายและมีประสิทธิภาพ

2.5.1 คุณสมบัติ

- ใช้ไมโครคอนโทรลเลอร์ ESP8266-12E ซึ่งมีหน่วยความจำ flash 4 MBytes
- มี GPIO 17 ขา , ADC 1 ขา
- ระดับสัญญาณลอจิกของสัญญาณอินพุตและเอาต์พุต (GPIO) 3.3 V

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- สามารถใช้ไฟเลี้ยงบอร์ดจากพอร์ต USB 5VDC หรือจากแหล่งจ่ายไฟภายนอก 5-10 VDC
- รองรับการเขียนโปรแกรมด้วย Lua และ Arduino IDE
- มีขนาดเล็ก ใช้งานง่าย
- ราคาถูก



รูปที่ 2.25 รูปบอร์ด NodeMCU V3 และขาต่างๆ

จากรูปที่ 2.25

- ขา INPUT/OUTPUT แบบ DIGITAL จะมีจำนวนขาใช้งานทั้งหมด 11 ขา ซึ่งสามารถกำหนดให้เป็นขา INPUT หรือขา OUTPUT ก็ได้ ตามการเขียนโปรแกรม โดยขาเหล่านี้จะทำงานที่ระดับแรงดัน 3.3 โวลต์ดีซี กระแสที่สามารถจ่ายและรับได้ 12 มิลลิแอมป์ นอกจากนี้ในบางขายังสามารถกำหนดฟังก์ชันพิเศษได้

- ขา INPUT แบบ ANALOG จะมีจำนวนขาใช้งาน 1 ขา (A0) ซึ่งในแต่ละขาจะมีระดับแรงดันสูงสุด 3.3 โวลต์ เมื่อเทียบกับกราวด์ และมีความละเอียดขนาด 10 บิต

Pin	GPIO	ฟังก์ชัน	รายละเอียด
D0	GPIO16	Wake up	ขณะ Boot จะมีสถานะเป็น HIGH
D1	GPIO5	SCL	I2C
D2	GPIO4	SDA	I2C
D3	GPIO0	FLASH	ต่ออยู่กับสวิตช์ FLASH, ไม่แนะนำให้สถานะเป็น LOW เพราะจะทำให้ไม่สามารถ Boot ได้
D4	GPIO2	BUILT-IN LED	ไม่แนะนำให้สถานะเป็น LOW เพราะจะทำให้ไม่สามารถ Boot ได้
D5	GPIO14	SCLK	SPI
D6	GPIO12	MISO	SPI
D7	GPIO13	MOSI	SPI
D8	GPIO15	CS	SPI, ไม่แนะนำให้สถานะเป็น HIGH เพราะจะทำให้ไม่สามารถ Boot ได้
RX	GPIO3	RX	อาจใช้งานเป็นขา Input ได้, ขณะ Boot จะมีสถานะเป็น HIGH
TX	GPIO1	TX	อาจใช้งานเป็นขา Input ได้, ขณะ Boot จะมีสถานะเป็น HIGH, ไม่แนะนำให้มีสถานะเป็น Low เพราะจะทำให้ไม่สามารถ Boot ได้
A0	ADC0	Input เท่านั้น	ขา Input แบบ Analog

ตารางที่ 2.4 ข้อมูลทางเทคนิค

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6 Google Sheet



รูปที่ 2.26 Google Sheet

Google Sheets หรือเรียกย่อ ๆ ว่า Sheets เป็นซอฟต์แวร์ด้าน Spreadsheet สร้างตารางคำนวณทำงานแบบ Online บน Cloud ใช้งานได้ฟรี ทำหน้าที่คล้าย Microsoft Excel เป็นตาราง ช่อง ใส่สูตรคำนวณได้ สามารถแชร์ให้กับคนอื่น เข้ามาทำงานร่วมกันได้ พร้อมแจ้งเตือนได้เมื่อมีการเปลี่ยนแปลงเอกสารทันที

2.6.1 ความสามารถของ Google Sheets

- สร้างตาราง สร้างการคำนวณมีสูตรคำนวณมากมาย
- สามารถจัดรูปแบบอัตโนมัติได้ คล้าย ๆ Conditional Formatting ใน Excel
- ทำการสรุปข้อมูลได้ด้วย Pivot Table
- มีกราฟให้เลือกมากมาย เช่น Column, Bar, Pie, Treemap, Map เป็นต้น
- ทำงานได้โดยใช้ Browser เท่านั้น ไม่ต้องติดตั้งโปรแกรมใด ๆ
- มีการบันทึกข้อมูลให้อัตโนมัติ (Autosave)
- แชร์ (Share) และแจ้งเตือน (Notification) ได้
- สามารถนำเอา Excel แปลงเป็น Sheets ได้
- สามารถ Download เอกสาร Sheets เป็นไฟล์ Excel, CSV, PDF ได้
- สามารถใช้เป็น Data Source ให้กับ Power BI หรือ Google Data Studio ได้
- สามารถติดตั้ง Add-Ons เพิ่มเติม เพื่อเพิ่มความสามารถให้ Sheets ได้ เช่น การทำ Mail Merge การนำเข้าข้อมูลจาก Google Classroom, Google Analytics เข้ามาใน Sheets ได้

2.7 App Script



รูปที่ 2.27 App Script

Apps Script คือ แพลตฟอร์มที่ช่วยในการพัฒนาแอปพลิเคชันด้วยการเขียนโค้ดที่น้อยลง (Low-Code) ซึ่งตอบโจทย์ในการขยายฟังก์ชันการทำงานใหม่ๆได้ด้วยตนเอง นอกจากนี้ยังสามารถปรับให้เป็นระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อัตโนมัติได้อย่างรวดเร็วและง่ายดาย โดยไม่จำเป็นต้องมีทักษะด้านการพัฒนาซอฟต์แวร์อย่างผู้เชี่ยวชาญแต่อย่างใด

Apps Script สามารถช่วยเปลี่ยนงานให้กลายเป็นระบบอัตโนมัติ โดยการเชื่อมต่อกับแอปพลิเคชันจาก Google Workspace ที่ใช้ในชีวิตประจำวัน โดยที่สามารถเพิ่มฟังก์ชันการทำงานหรือปรับแต่งการทำงานได้ทุกที่ทุกเวลาที่จำเป็น ทำให้ผู้ใช้มีเวลาให้ความสำคัญกับงานส่วนอื่น และยังสามารถใช้เวลาให้เกิดประโยชน์สูงสุดอีกด้วย นอกจากนี้ Apps Script สามารถใช้งานได้หลากหลาย เช่น

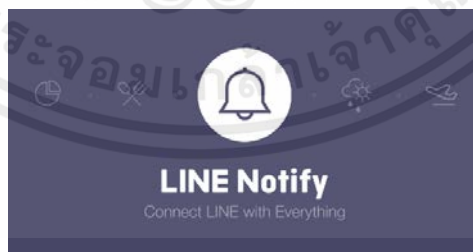
- สามารถเพิ่มเมนู กล้องโต้ตอบ และแถบด้านข้างที่ต้องการใน Google Docs, Sheets, และ Forms ได้ด้วยตนเอง
- สามารถเขียน functions และ macros ใน Google Sheets ด้วยตนเอง
- สามารถเผยแพร่เว็บแอป ทั้งแบบ Standalone หรือแบบฝังใน Google Sites
- สามารถโต้ตอบกับบริการอื่นๆ ของ Google ได้อย่างรวดเร็วและง่ายดาย ไม่ว่าจะเป็น AdSense, Analytics, Calendar, Drive, Gmail และ Maps
- สามารถสร้างส่วนเสริมและเผยแพร่ไปยัง Google Workspace Marketplace

2.7.1 วิธีการสร้าง Script

- หากใช้ Docs, Sheets, หรือ Slides ให้คลิกที่เครื่องมือ (Tools) จากนั้นเลือกโปรแกรมแก้ไขสคริปต์ (Script editor)
- หากใช้ Forms ให้คลิกเพิ่มเติม (More) จากนั้นเลือกโปรแกรมแก้ไขสคริปต์ (Script editor)
- สร้างสคริปต์

Apps Script รองรับภาษายอดนิยมบนเว็บ อาทิ HTML, CSS และ JavaScript ทำให้คุณสามารถสร้างงานได้โดยไม่ต้องเรียนรู้เฟรมเวิร์กใหม่

2.8 LINE Notify



รูปที่ 2.28 Line Notify

Line Notify คือ บริการที่ Line ได้เตรียมไว้ให้ในรูปแบบของ API ให้กับนักพัฒนา สามารถนำไปใช้ต่อยอด พัฒนาโปรเจกต์ ที่มีความต้องการส่งข้อความในการแจ้งเตือนเข้าไปยัง บัญชีส่วนตัวของเรา หรือ กลุ่มได้ สามารถรับข้อความแจ้งเตือนจากเว็บเซอร์วิสต่าง ๆ ที่สนใจได้ทาง LINE โดยหลังเสร็จสิ้นการเชื่อมต่อกับทางเว็บเซอร์วิสแล้วจะได้รับการแจ้งเตือนจากบัญชีทางการของ “LINE Notify” ซึ่งให้บริการโดย LINE โดยบริการหลักๆ ที่สามารถเชื่อมต่อได้แก่ GitHub, IFTTT หรือ Mackerel เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

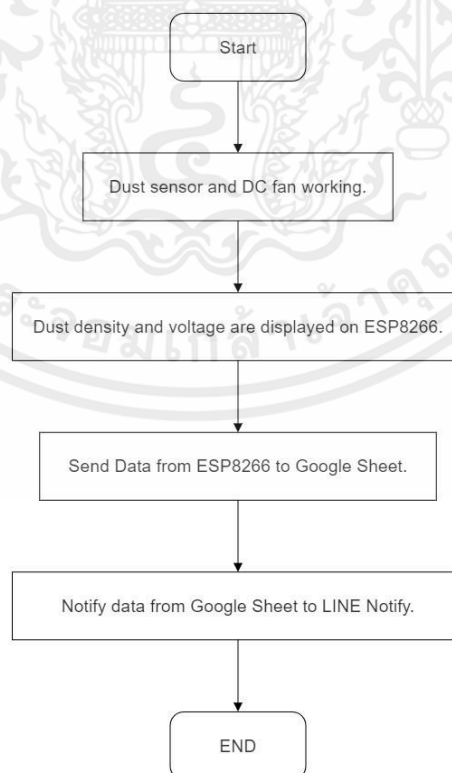
หลักการงานและการออกแบบ

3.1 วัสดุอุปกรณ์ เครื่องมือ โปรแกรมและแอปพลิเคชัน

- เครื่องกรองอากาศ
- พัดลมกระแสตรง (Direct Current Fan : DC Fan)
- Wemos D1 Mini - ESP8266
- NodeMCU ESP8266 (NodeMCU V3)
- Optical Dust Sensor PM2.5 (GP2Y1014AU0F)
- สาย Micro USB
- โปรแกรม Arduino IDE
- Google Sheets , App Script
- Application Line
- Line Notify

3.2 แผนผังการทำงาน

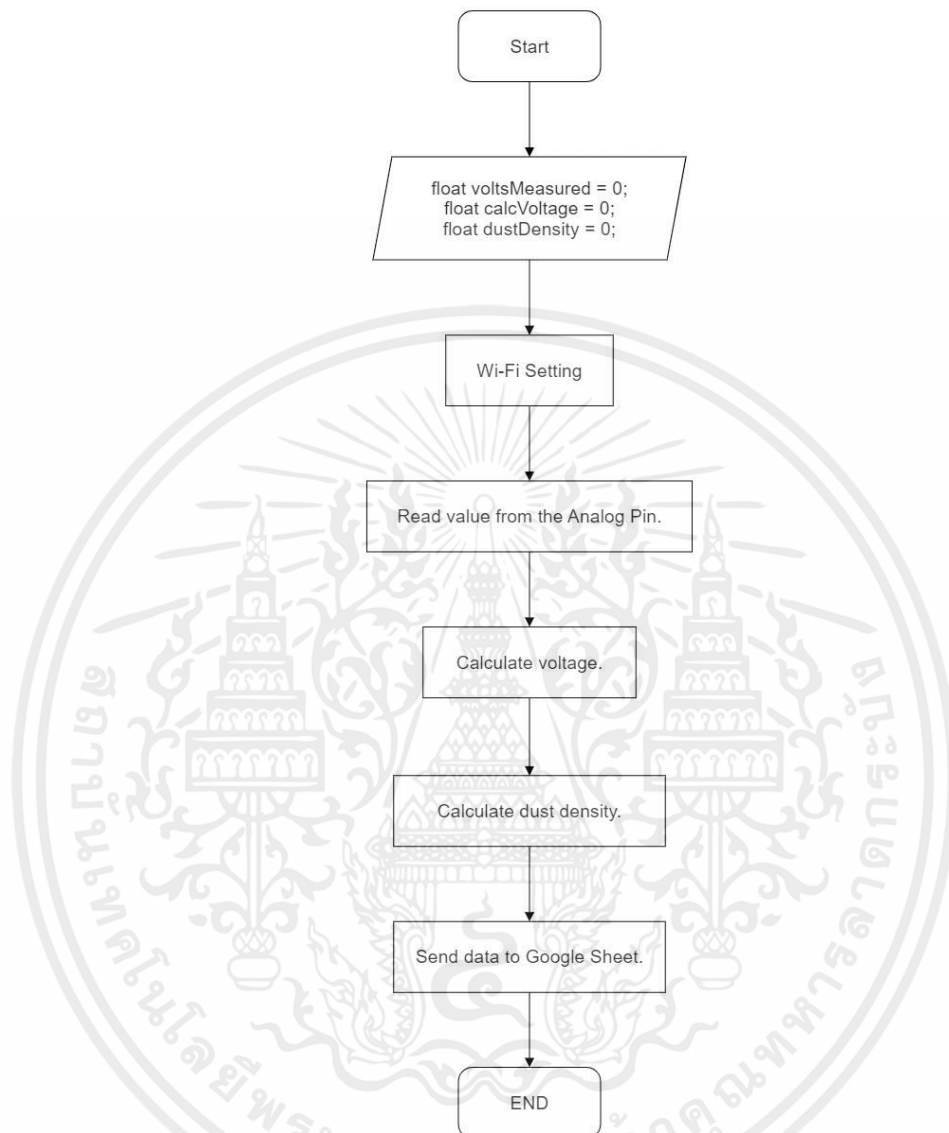
3.2.1 การทำงานของระบบ



รูปที่ 3.1 แผนภาพ Flowchart การทำงานของระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

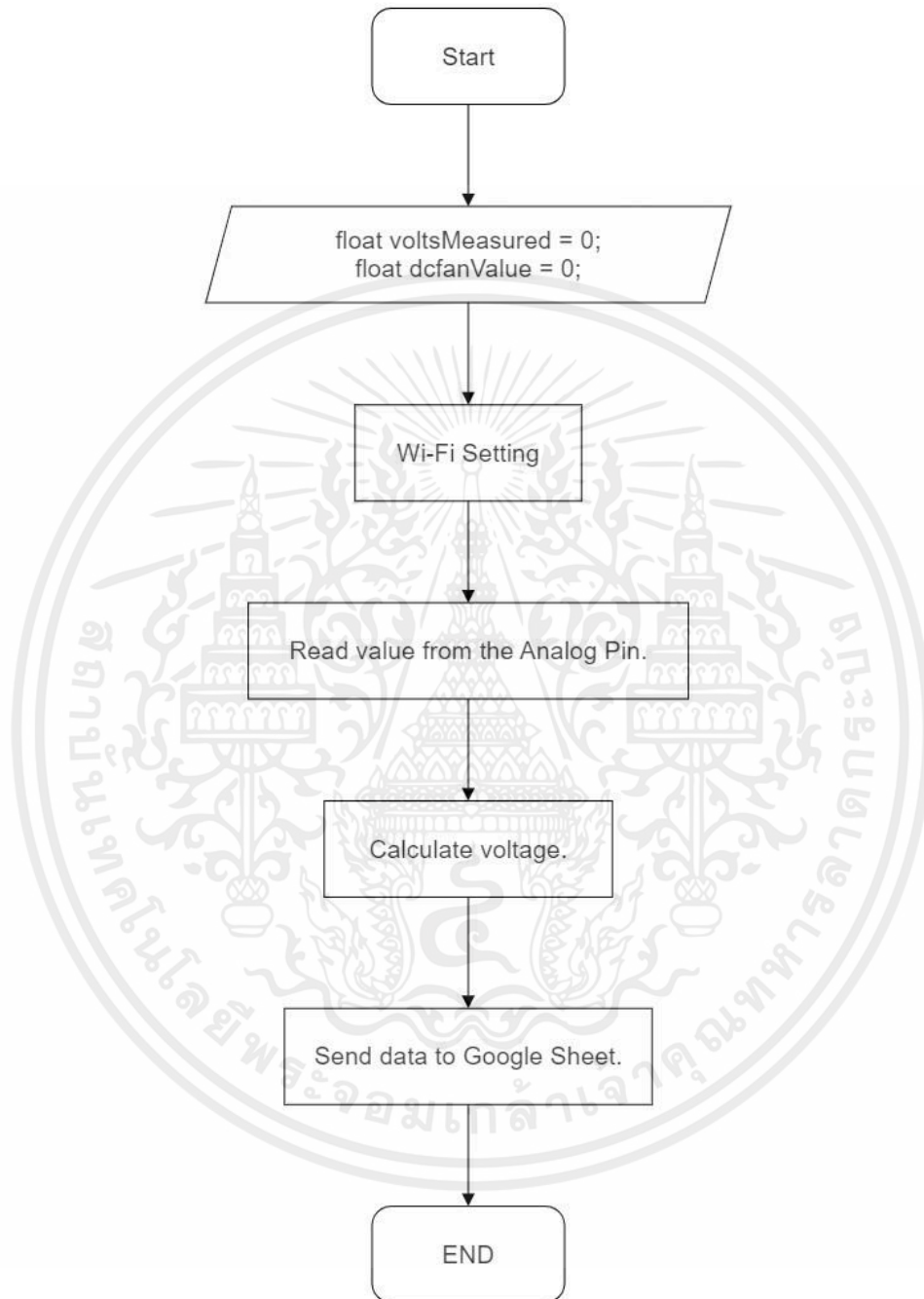
3.2.2 การทำงานของ Wemos D1 Mini - ESP8266 ร่วมกับ Optical Dust Sensor PM2.5 (GP2Y1014AU0F)



รูปที่ 3.2 แผนภาพ Flowchart การทำงานของ Wemos D1 Mini - ESP8266 ร่วมกับ Optical Dust Sensor PM2.5 (GP2Y1014AU0F)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

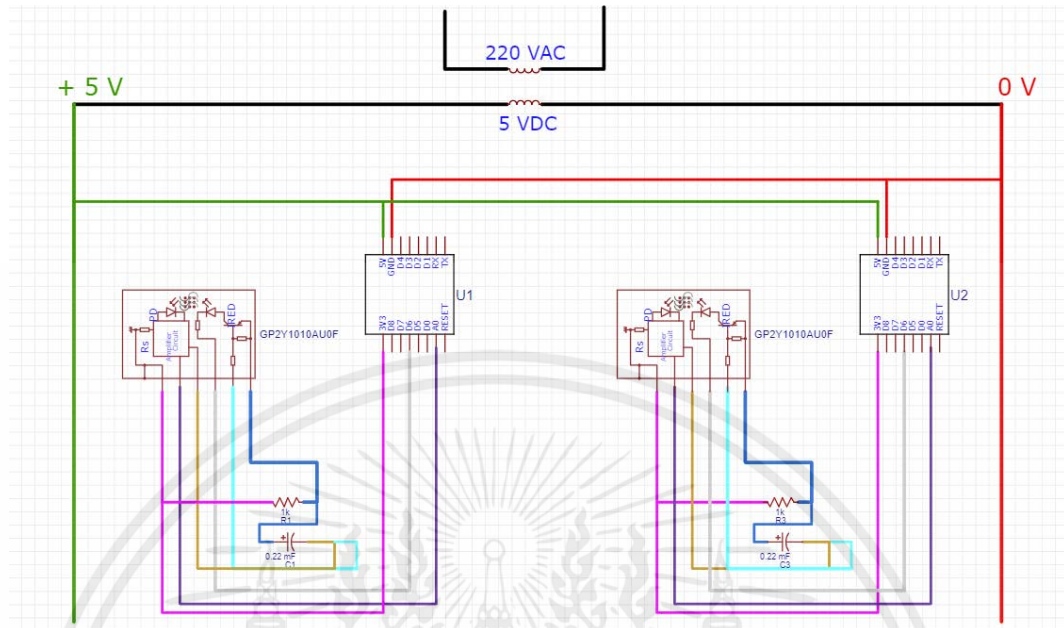
3.2.3 การทำงานของ NodeMCU ESP8266 (NodeMCU V3) ร่วมกับพัดลมกระแสตรง (Direct Current Fan : DC Fan)



รูปที่ 3.3 แผนภาพ Flowchart การทำงานของ NodeMCU ESP8266 (NodeMCU V3) ร่วมกับพัดลมกระแสตรง (Direct Current Fan : DC Fan)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 แผนผังของระบบ



รูปที่ 3.4 Schematic ของ ระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

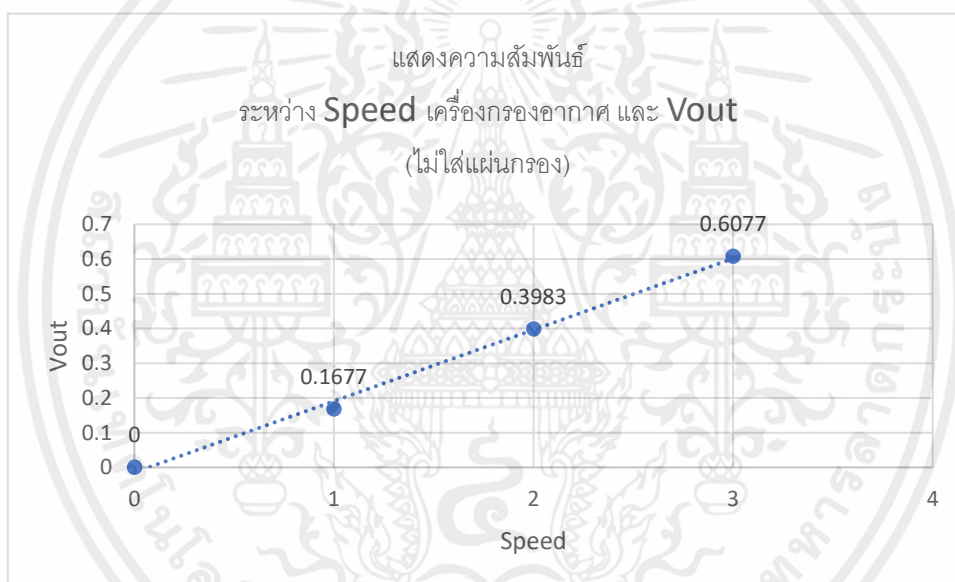
บทที่ 4

ผลการทดลอง

4.1 หาความสัมพันธ์ระหว่าง speed เครื่องกรองอากาศและ Vout

ตารางที่ 4.1 วัดแรงดัน DC Fan ที่ติดตั้งภายในเครื่องกรองอากาศ (ไม่ใส่แผ่นกรอง)

speed	แรงดัน (Volt)	แรงดัน (Volt)	แรงดัน (Volt)	แรงดันเฉลี่ย (Volt)
0	0	0	0	0
1	0.609	0.606	0.608	0.1677
2	0.395	0.398	0.402	0.3983
3	0.166	0.168	0.169	0.6077

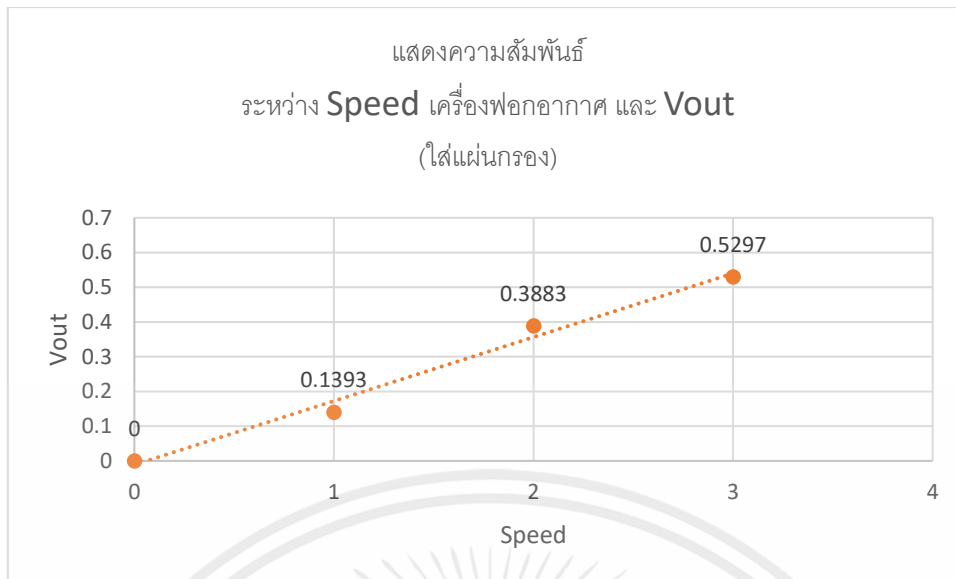


รูปที่ 4.1 กราฟแสดงความสัมพันธ์ระหว่าง Speed เครื่องกรองอากาศ และ V_{out} (ไม่ใส่แผ่นกรอง)

ตารางที่ 4.2 วัดแรงดัน DC Fan ที่ติดตั้งภายในเครื่องกรองอากาศ (ใส่แผ่นกรอง : HEPA + Carbon)

speed	แรงดัน (Volt)	แรงดัน (Volt)	แรงดัน (Volt)	แรงดันเฉลี่ย (Volt)
0	0	0	0	0
1	0.528	0.532	0.529	0.1393
2	0.388	0.385	0.392	0.3883
3	0.137	0.140	0.141	0.5297

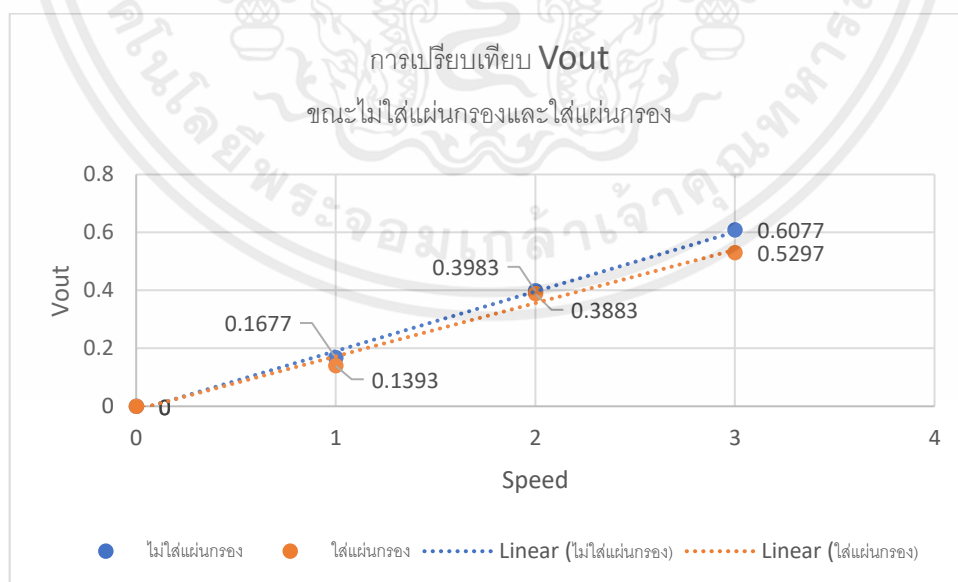
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.2 กราฟแสดงความสัมพันธ์ระหว่าง Speed เครื่องกรองอากาศ และ V_{out} (ใส่แผ่นกรอง)

ตารางที่ 4.3 เปอร์เซ็นต์แรงดันที่ลดลง

Speed	แรงดันขณะไม่มีใส่แผ่นกรอง (Volt)	แรงดันขณะใส่แรงดัน (Volt)	เปอร์เซ็นต์แรงดันลดลง
0	0	0	0
1	0.1677	0.1393	0.1694 %
2	0.3983	0.3883	0.0251 %
3	0.6077	0.5297	0.1284 %

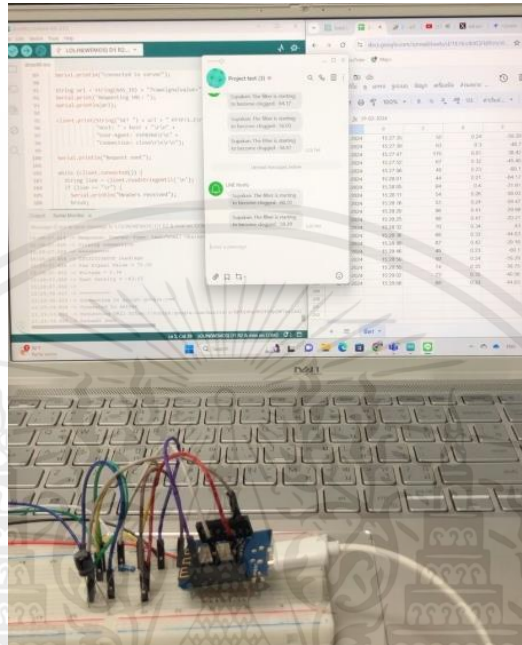


รูปที่ 4.3 กราฟการเปรียบเทียบ V_{out} ขณะไม่มีใส่แผ่นกรองและใส่แผ่นกรอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 การเขียนโปรแกรมและผลลัพธ์การแสดงผล

ทำการจ่ายไฟเข้าที่เครื่องกรองอากาศ และเขียนโปรแกรมที่ Arduino IDE เพื่อให้ Dust Sensor ส่งข้อมูลฝุ่น ที่ตรวจจับได้มาที่ ESP8266 จากนั้น ESP8266 จะทำการประมวลผลค่าความหนาแน่นของฝุ่นแล้วแสดงผลที่ ESP8266 และ ESP8266 จะทำการส่งข้อมูลที่แสดงผลอยู่ไปยัง Google Sheet



รูปที่ 4.1 ตัวอย่าง เขียนโปรแกรม Arduino IDE และ Google Sheet รวมไปถึงการแจ้งเตือน LINE Notify

๕.๕

```
// แจ้งเตือนผ่านไลน์เมื่อ column C (Outlet dust density) มีค่าน้อยกว่า 420
if (parseInt(rowData[3]) < 420) {
  var message = 'The filter is starting to become clogged: ' + rowData[3];
  sendLineNotification(message);
}
```

รูปที่ 4.2 เงื่อนไขให้แจ้งเตือนที่ LINE Application ผ่าน LINE Notify

๕.๕

จากนั้นเขียนโปรแกรม Apps Script เพื่อให้มีการแจ้งเตือนที่ LINE Application ผ่าน LINE Notify ตามเงื่อนไขที่กำหนดไว้ โดยตัวอย่างจะเป็นการกำหนดเงื่อนไขให้มีการแจ้งเตือนเมื่อค่าความหนาแน่นของฝุ่นน้อยกว่า 420

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

09-03-2024	19:04:02		415.4
09-03-2024	19:04:03	509.04	
09-03-2024	19:04:06		415.4
09-03-2024	19:04:07	509.04	
09-03-2024	19:04:11		414.59
09-03-2024	19:04:12	509.04	
09-03-2024	19:04:15		414.59
09-03-2024	19:04:16	509.04	
09-03-2024	19:04:20		414.59
09-03-2024	19:04:24	509.04	
09-03-2024	19:04:25		414.59
09-03-2024	19:04:29	509.04	
09-03-2024	19:04:29		415.4
09-03-2024	19:04:33	509.04	
09-03-2024	19:04:34		417.85
09-03-2024	19:04:37	509.04	
09-03-2024	19:04:39		424.36
09-03-2024	19:04:41	509.04	
09-03-2024	19:04:43		430.06
09-03-2024	19:04:45	509.04	
09-03-2024	19:04:47		432.5
09-03-2024	19:04:50	509.04	
09-03-2024	19:04:51		432.5

รูปที่ 4.3 แสดงข้อมูลที่ได้ที่ Google Sheet

๔.๖

Project test (3)

LINE Notify

Supakan: The filter is starting to become clogged: 415.40

Supakan: The filter is starting to become clogged: 415.40

Supakan: The filter is starting to become clogged: 414.59

Supakan: The filter is starting to become clogged: 414.59

Supakan: The filter is starting to become clogged: 414.59

Supakan: The filter is starting to become clogged: 414.59

Supakan: The filter is starting to become clogged: 415.40

Supakan: The filter is starting to become clogged: 417.85

7:04 PM

รูปที่ 4.4 ข้อมูลที่แจ้งเตือนจากเงื่อนไขที่ให้มีการแจ้งเตือนผ่าน LINE Notify เมื่อค่าความหนาแน่น

น้อยกว่า 420

๔.๗

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปผลการทดลอง

5.1 สรุปผลการทดลอง

จากการทำโครงการเรื่องการควบคุมความเร็วพัดลมดูดฝุ่นเครื่องกรองอากาศ ได้มีการศึกษา และ ออกแบบเพื่อให้เครื่องกรองอากาศ สามารถตรวจสอบค่าฝุ่นขาเข้า ฝุ่นขาออก และแจ้งเตือนประสิทธิภาพของ แผ่นกรองอากาศ โดยการเขียนคำสั่งให้ Wemos D1 Mini - ESP8266 ทำการรับค่าอินพุตจาก Optical Dust Sensor PM2.5 (GP2Y1014AU0F) นำไปบันทึกผลที่ Google Sheets โดยเขียนคำสั่งผ่าน Arduino IDE และ App Script และสามารถแจ้งเตือนข้อมูลผ่าน Application Line ด้วย Line Notify

5.2 วิจารณ์ผลการทดลอง

จากการทำโครงการเรื่องการควบคุมความเร็วพัดลมดูดฝุ่นเครื่องกรองอากาศ ผลการทดลองของ โครงการมีความคลาดเคลื่อน เนื่องจากอุปกรณ์มีประสิทธิภาพไม่เพียงพอ อุปกรณ์มีความคลาดเคลื่อนในตัว ของอุปกรณ์เอง สภาพอากาศและบริเวณที่ใช้ในการทดลองไม่เหมาะสม กล่าวคือ ห้องทดลองไม่ใช่ห้องปิด

5.3 ข้อเสนอแนะ

5.3.1 การทดลองมีปัญหาเกี่ยวกับการเขียนคำสั่งให้กับฮาร์ดแวร์ทำงาน ควรศึกษาการทำงานของ ฮาร์ดแวร์ให้เข้าใจ เพื่อไม่ให้เกิดความผิดพลาดในการเขียนโปรแกรมคำสั่ง

5.3.2 การทดลองมีปัญหาเกี่ยวกับการจ่ายไฟ เมื่อทำการติดตั้งอุปกรณ์พบปัญหาอุปกรณ์ไม่สามารถ ทำงานได้ ควรศึกษาการจ่ายไฟให้ดีกว่าก่อนลงมือทดลอง

5.3.3 การออกแบบ และการใช้วัสดุอุปกรณ์ในการติดตั้งมีผลต่อความปลอดภัยของผู้ทดลองและ ผู้ใช้งาน

เอกสารอ้างอิง

- [1] Tech&Time system Co., Ltd. สืบค้นเมื่อ 1 ตุลาคม 2566
จาก <https://www.tech-time.co.th/>
- [2] EuroVent. สืบค้นเมื่อ 1 ตุลาคม 2566
จาก https://www.eurovent.co.th/index.php?route=extension/extension/blog/article&article_id=5&category=1
- [3] มอเตอร์ไฟฟ้ากระแสตรง. สืบค้นเมื่อ 1 ตุลาคม 2566
จาก <https://www.prd.go.th/th/file/get/file/20210729d41d8cd98f00b204e9800998ecf8427e140258.pdf>
- [4] iToolmart. สืบค้นเมื่อ 1 ตุลาคม 2566
จาก <https://itoolmart.com/blog/content/7dqjr>
- [5] The invention. สืบค้นเมื่อ 2 ตุลาคม 2566
จาก <https://www.ai-corporation.net/2021/11/16/what-is-a-motor/>
- [6] Misumitechnical. สืบค้นเมื่อ 2 ตุลาคม 2566
จาก <https://misumitechnical.com/technical/mechanical/type-of-industrial-motor/>
- [7] TMO. สืบค้นเมื่อ 2 ตุลาคม 2566
จาก <https://toolmartonline.com/powertools/>
- [8] Phelps Dodge Thailand. สืบค้นเมื่อ 10 ตุลาคม 2566
จาก <https://www.pdcable.com/>
- [9] JL Home. สืบค้นเมื่อ 10 ตุลาคม 2566
จาก https://www.jlhome.in.th/index.php?route=simple_blog/article/view&simple_blog_article_id=48
- [10] dwacool. สืบค้นเมื่อ 10 ตุลาคม 2566
จาก <https://www.dwacool.com/what-do-different-types-of-filters-do/>
- [11] TFS. สืบค้นเมื่อ 10 ตุลาคม 2566
จาก <https://www.thaifiltersupply.com/news/Detail/0-12->
- [12] VOV International Co., Ltd.(HQ). สืบค้นเมื่อ 10 ตุลาคม 2566
จาก <https://www.vovgroup.com/blogs-cleanroom-what-is-hepa-filter-why-important-in-cleanroom>
- [13] Youtube. สืบค้นเมื่อ 10 ตุลาคม 2566
จาก https://www.youtube.com/watch?v=gxC1cw2_S_I&t=545s

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

[14] เซ็นเซอร์วัดฝุ่น PM2.5 Keyestudio GP2Y1014AU สืบค้นเมื่อ 5 พฤศจิกายน 2566

จาก <https://www.robotsiam.com/product/238/%E0%B9%80%E0%B8%8B%E0%B9%87%E0%B8%99%E0%B9%80%E0%B8%8B%E0%B8%AD%E0%B8%A3%E0%B9%8C%E0%B8%A7%E0%B8%B1%E0%B8%94%E0%B8%9D%E0%B8%B8%E0%B9%88%E0%B8%99-pm2-5-keyestudio-gp2y1014au>

[15] PM2.5 GP2Y1014AU0F Replace GP2Y1010AU0F Optical Dust Sensor PM2.5

สืบค้นเมื่อ 15 พฤศจิกายน 2566

จาก https://shopee.co.th/product/603297687/20395125083?d_id=c7123&uls_trackid=4uvbt8f400lj&utm_content=2rYTHrLcGG12hUC4hmFZITtEUe9M

[16] How Does the GP2Y1014AU0F Dust Sensor Work and How to Interface it with Arduino

สืบค้นเมื่อ 17 พฤศจิกายน 2567

จาก <https://circuitdigest.com/microcontroller-projects/interfacing-dust-sensor-with-arduino>

[17] GP2Y1014AU0F_SPEC-1288381 สืบค้นเมื่อ 25 พฤศจิกายน 2566

ภาคผนวก

```
#include <ESP8266WiFi.h>
```

```
#include <WiFiClientSecure.h>
```

```
const char* ssid = "kycrazy"; // ชื่อของ Hotspot
```

```
const char* password = "s192002191919s"; // รหัสผ่านของ Hotspot
```

```
const char* host = "script.google.com";
```

```
const int httpsPort = 443;
```

```
WiFiClientSecure client;
```

```
long now = millis();
```

```
long lastMeasure = 0;
```

```
const char* GAS_ID =
```

```
"https://script.google.com/macros/s/AKfycby5MjTT8pDW7sxlkkLdDT6t76j7g4C-  
pVbHA_JUGWVeWMfOLkWB8wmRQTPE5IPFjIP/exec";
```

```
const int dustPin = A0; // dust sensor - Wemos A0 pin
```

```
const int ledPin = D6;
```

```
float voltsMeasured = 0;
```

```
float calcVoltage = 0;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

float dustDensity = 0;

void setup() {
  Serial.begin(115200); // เริ่มต้นการสื่อสารผ่าน Serial Monitor
  pinMode(ledPin, OUTPUT);

  // Connect to Hotspot
  Serial.println();
  Serial.println("Connecting to Hotspot...");

  WiFi.begin(ssid, password); // เริ่มต้นการเชื่อมต่อ Hotspot
  Serial.println("");

  while (WiFi.status() != WL_CONNECTED) { // รอจนกว่าจะเชื่อมต่อสำเร็จ
    delay(500);
    Serial.print(".");
  }

  Serial.println("");
  Serial.println("Hotspot connected");

  Serial.print("IP Address: ");
  Serial.println(WiFi.localIP()); // แสดง IP address ที่ได้จากการเชื่อมต่อ Hotspot

  client.setInsecure();
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

void loop() {

  digitalWrite(ledPin, LOW); // เปิดไฟ LED

  delayMicroseconds(280);

  voltsMeasured = analogRead(dustPin); // อ่านค่าจากเซ็นเซอร์ฝุ่น

  delayMicroseconds(40);

  digitalWrite(ledPin, HIGH); // ปิดไฟ LED

  delayMicroseconds(9680);

  now = millis();
  if (now - lastMeasure > 1000) {
    lastMeasure = now;

    calcVoltage = voltsMeasured * (3.3 / 689.0); // คำนวณค่าแรงดัน
    dustDensity = (0.17 * calcVoltage - 0.1) * 1000 ; // คำนวณความหนาแน่นของฝุ่น (สูงสุด 689 3.3
    461 & 666 3.19 442.27)

    Serial.println("GP2Y1010AU0F readings");
    Serial.print("Raw Signal Value = ");
    Serial.println(voltsMeasured);
    Serial.print("Voltage = ");
    Serial.println(calcVoltage);
    Serial.print("Dust Density = "); // ug/m3

    Serial.println(dustDensity); // แสดงผลลัพธ์ทาง Serial Monitor
  }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Serial.println("");

delay(1000); // รอเวลา

sendData(voltsMeasured, calcVoltage, dustDensity);
}
}

void sendData(float voltsMeasured, float calcVoltage, float dustDensity) {
  Serial.println("=====");
  Serial.print("Connecting to ");
  Serial.println(host);

  if (!client.connect(host, httpsPort)) {
    Serial.println("Connection failed");
    return;
  }

  Serial.println("Connected to server");

  String url = String(GAS_ID) + "?rawsignalvalue=" + String(voltsMeasured) + "&voltage=" +
String(calcVoltage) + "&dustdensity=" + String(dustDensity);

  Serial.print("Requesting URL: ");

  Serial.println(url);

  client.print(String("GET ") + url + " HTTP/1.1\r\n" +
  เอกสารนี้เป็นเอกสารสงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
  ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

```

"Host: " + host + "\r\n" +
"User-Agent: ESP8266\r\n" +
"Connection: close\r\n\r\n");

```

```
Serial.println("Request sent");
```

```
while (client.connected()) {
```

```
String line = client.readStringUntil('\n');
```

```
if (line == "\r") {
```

```
Serial.println("Headers received");
```

```
break;
```

```
}
```

```
}
```

```
String line = client.readStringUntil('\n');
```

```
if (line.startsWith("{\"state\":\"success\""})) {
```

```
Serial.println("ESP8266/Arduino successfully sent data to Google Sheet");
```

```
} else {
```

```
Serial.println("ESP8266/Arduino failed to send data to Google Sheet");
```

```
}
```

```
Serial.print("Response: ");
```

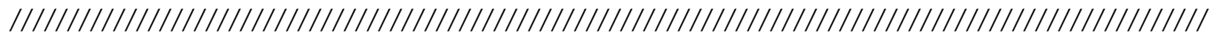
```
Serial.println(line);
```

```
Serial.println("Closing connection");
```

```
Serial.println("=====");
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

}



----- (END OF ARDUINO CODE) -----



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

function doGet(e) {

    Logger.log( JSON.stringify(e) );

    var result = 'Ok';

    if (e.parameter == 'undefined') {

        result = 'No Parameters';

    }

    else {

        var sheet_id = '1tLHcvtbXQFkBRmcVi1YYP6LWDXxLecam1Dta7R1Y-VI'; // Spreadsheet
ID
        var sheet = SpreadsheetApp.openById(sheet_id).getActiveSheet();
        var newRow = sheet.getLastRow() + 1;
        var rowData = [];
        var Curr_Date = new Date();
        rowData[0] = Utilities.formatDate(Curr_Date, "Asia/Bangkok", 'dd-MM-yyyy'); // Date in
column A
        var Curr_Time = new Date();
        rowData[1] = Utilities.formatDate(Curr_Time, "Asia/Bangkok", 'HH:mm:ss'); // Time in column
B

        for (var param in e.parameter) {

            Logger.log('In for loop, param=' + param);

            var value = stripQuotes(e.parameter[param]);

            Logger.log(param + ':' + e.parameter[param]);

            switch (param) {

                case 'rawsignalvalue':

                    rowData[2] = value; // Raw Signal Value in column C

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        result = 'Raw Signal Value Written on column C';

        break;

    case 'voltage':

        rowData[3] = value; // Voltage in column D

        result = 'Voltage Written on column D';

        break;

    case 'dustdensity':

        rowData[4] = value; // Dust Density in column E

        result = 'Dust Density Written on column E';

        break;

    }

}

Logger.log(JSON.stringify(rowData));

var newRange = sheet.getRange(newRow, 1, 1, rowData.length);
newRange.setValues([rowData]);

// แจ้งเตือนผ่านไลน์เมื่อ column E มีค่าน้อยกว่า -50
if (parseInt(rowData[4]) < -50) {

    var message = 'The filter is starting to become clogged: ' + rowData[4];

    sendLineNotification(message);

}

}

return ContentService.createTextOutput(result);

}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
function stripQuotes( value ) {  
  
    return value.replace(/^[\"']|["']$/g, "");  
  
}  
  
function sendLineNotification(message) {  
  
    var token = 'olPYsytlbpDtlyxuXiZYntq1Vrt6EgzJpJPnYKYklax'; // ใส่ token ที่คุณได้รับจาก LINE  
Notify ที่นี่  
  
    var url = 'https://notify-api.line.me/api/notify';  
  
    var headers = {  
  
        'Authorization': 'Bearer ' + token  
  
    };  
  
    var payload = {  
  
        'message': message  
  
    };  
  
    var options = {  
  
        'method': 'post',  
  
        'headers': headers,  
  
        'payload': payload  
  
    };  
  
    var response = UrlFetchApp.fetch(url, options);  
  
    Logger.log(response.getContentText());  
  
}  
  
////////////////////////////////////  
----- (END OF APPS SCRIPT) -----
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้