



โครงการ

ระบบตรวจจับป้องกันฝุ่น PM2.5 และแจ้งเตือนอัตโนมัติ

ชนาธิป รวงผึ้ง

64160086

โครงการนี้เป็นส่วนหนึ่งของการศึกษาหลักสูตรปริญญาวิทยาศาสตรบัณฑิต

สาขาวิชาเทคโนโลยีสารสนเทศเพื่ออุตสาหกรรมดิจิทัล

คณะวิทยาการสารสนเทศมหาวิทยาลัยบูรพา

ปีการศึกษา 2567

ลิขสิทธิ์ของคณะวิทยาการสารสนเทศ มหาวิทยาลัยบูรพา



PROJECT

THE PM2.5 DUST DETECTION PREVENTION SYSTEM AND AUTOMATIC NOTIFICATION

CHANATHIP RUANGPHUENG

64160086

A PROJECT SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENT
FOR THE BACHELOR DEGREE OF SCIENCE IN INFORMATION
TECHNOLOGY FOR DIGITAL INDUSTRY
FACULTY OF INFORMATICS BURAPHA UNIVERSITY
ACADEMIC YEAR 2024.

COPYRIGHT FACULTY OF INFORMATICS BURAPHA UNIVERSITY



คณะวิทยาการสารสนเทศ มหาวิทยาลัยบูรพา

ใบรับรองโครงการ

หัวข้อโครงการ/รายงาน ระบบตรวจจับป้องกันฝุ่น PM2.5 และแจ้งเตือนอัตโนมัติ

THE PM2.5 DUST DETECTION PREVENTION SYSTEM AND AUTOMATIC
NOTIFICATION

ชื่อนิสิต

นายชนาธิป รวงผึ้ง

รหัสประจำตัว 64160086

อาจารย์ที่ปรึกษา

อาจารย์สิทธิศักดิ์ แซ่จิ่ง

วันที่สอบ

19 ธันวาคม 2567

โครงการนี้ได้ผ่านการเห็นชอบจากคณะกรรมการสอบ

ให้เป็นโครงการหลักสูตรวิทยาศาสตรบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศเพื่อ
อุตสาหกรรมดิจิทัล

(ดร.กามาล บาชะ)

ประธานกรรมการ

(อาจารย์สิทธิศักดิ์ แซ่จิ่ง)

กรรมการ

หัวข้อโครงการ	ระบบตรวจจับป้องกันฝุ่น PM2.5 และแจ้งเตือนอัตโนมัติ
นิสิต	นายชนาธิป รวงผึ้ง
รหัสประจำตัว	64160086
อาจารย์ที่ปรึกษา	อาจารย์สิทธิศักดิ์ แซ่จิ่ง
ระดับการศึกษา	วิทยาศาสตร์บัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ เทคโนโลยีสารสนเทศเพื่ออุตสาหกรรมดิจิทัล
คณะ	วิทยาการสารสนเทศ มหาวิทยาลัยบูรพา
ปีการศึกษา	2567

บทคัดย่อ

โครงการนี้มีวัตถุประสงค์เพื่อพัฒนาระบบตรวจจับและแสดงผลค่าฝุ่น PM2.5 ด้วยเทคโนโลยี Internet of Things (IoT) เพื่อช่วยในการตรวจสอบคุณภาพอากาศอย่างสะดวกและรวดเร็ว ระบบประกอบด้วย NodeMCU V2 ซึ่งเป็นไมโครคอนโทรลเลอร์ที่เชื่อมต่อกับเซ็นเซอร์ตรวจวัดฝุ่น PMS5003 G7 และจอ LCD สำหรับแสดงค่าฝุ่น PM2.5 แบบเรียลไทม์ ระบบทำงานโดยการอ่านค่าฝุ่นจากเซ็นเซอร์ PMS5003 G7 ซึ่งสามารถวัดค่าฝุ่นละอองขนาดเล็กได้อย่างแม่นยำ จากนั้นจะแสดงค่าฝุ่นที่ตรวจจับได้บนจอ LCD ผู้ใช้งานสามารถติดตามสถานะคุณภาพอากาศได้ตลอดเวลา โดยอุปกรณ์ทั้งหมดได้รับการออกแบบให้มีขนาดกะทัดรัดและใช้งานง่ายผลการทดสอบระบบแสดงให้เห็นว่าระบบสามารถตรวจจับค่าฝุ่น PM2.5 ได้อย่างถูกต้องและแสดงผลได้อย่างรวดเร็ว เหมาะสำหรับการใช้งานในครัวเรือนหรือพื้นที่ที่ต้องการตรวจสอบคุณภาพอากาศอย่างต่อเนื่อง โดยระบบนี้สามารถช่วยสร้างความตระหนักรู้ในเรื่องคุณภาพอากาศและลดความเสี่ยงต่อสุขภาพจากปัญหาฝุ่น PM2.5 ได้อย่างมีประสิทธิภาพ

Project Title	The PM2.5 Dust Detection Prevention System and Automactic Notification
Student	Mr. Chanathip Ruangphueng
Student ID	64160086
Advisor	Sittisak SaeChueng
Level of Study	Bachelor degree of Science in Information Technology For Digital Industry
Faculty	Faculty of Informatics, Burapha University
Year	2024

Abstract

This project aims to develop a The PM2.5 Aetection and Display System using Internet of Things (IoT) technology to facilitate convenient and real-time air quality monitoring. The system consists of a NodeMCU V2 microcontroller connected to a PMS5003 G7 dust sensor and an LCD screen for displaying PM2.5 levels in real-time. The system operates by reading air quality data from the PMS5003 G7 sensor, which accurately measures fine particulate matter (PM2.5). The detected values are then displayed on the LCD screen, allowing users to monitor air quality instantly. The hardware is designed to be compact and user-friendly for ease of use. Testing results indicate that the system can effectively detect PM2.5 levels with accuracy and display the information promptly. This system is suitable for household use or in areas requiring continuous air quality monitoring. It raises awareness about air quality issues and helps mitigate health risks caused by PM2.5 pollution effectively.

กิตติกรรมประกาศ

โครงการการออกแบบและพัฒนาระบบตรวจจับป้องกันฝุ่นPM2.5และแจ้งเตือนอัตโนมัติจัดทำเพื่อ
การสำเร็จการศึกษาของนักศึกษาระดับปริญญาตรี สามารถดำเนินการจนประสบความสำเร็จลุล่วงไป
ด้วยดี เนื่องจากได้รับความอนุเคราะห์และสนับสนุนเป็นอย่างดีจาก อาจารย์ สิทธิศักดิ์ แซ่จิ่งอาจารย์ที่
ปรึกษา ที่ได้กรุณาให้คำปรึกษา ความรู้ ข้อคิด ข้อเสนอแนะ และปรับปรุงแก้ไขข้อบกพร่องต่าง ๆ จนกระทั่ง
การวิจัยครั้งนี้สำเร็จเรียบร้อยด้วยดี ขอกราบขอบพระคุณอาจารย์เป็นอย่างสูง

ขอบคุณอาจารย์ อาจารย์สิทธิศักดิ์ แซ่จิ่ง ที่คอยติดตามงาน ให้คำปรึกษา แก้ไข และ คอยให้
คำแนะนำเกี่ยวกับการทำโครงการตลอดระยะเวลาการทำโครงการจนสำเร็จลุล่วงไปด้วยดีขอขอบคุณ
เพื่อนร่วมงานทุกคนที่อำนวยความสะดวกและช่วยเหลือในการทำวิจัยครั้งนี้

ขอบคุณอาจารย์คณะวิทยาการสารสนเทศ สาขาเทคโนโลยีสารสนเทศเพื่ออุตสาหกรรมดิจิทัล ที่
ให้ความรู้ในรายวิชาต่าง ๆ จนทำให้มีความรู้และความเข้าใจ จนสามารถนำความรู้ที่ได้มาประยุกต์ใช้กับ
โครงการนี้จนลุล่วงตามเป้าหมายที่หวังไว้

สุดท้ายนี้ผู้จัดทำโครงการหวังว่างานวิจัยฉบับนี้คงเป็นประโยชน์สำหรับทุกคน และผู้ที่สนใจ
ศึกษาต่อไป ผู้จัดทำโครงการขอขอบคุณทุกท่านที่มีส่วนเกี่ยวข้องทำให้โครงการเล่มนี้เสร็จสมบูรณ์
ขอขอบพระคุณอย่างสูงไว้ ณ ที่นี้

นายชนาธิป รวงผึ้ง

ธันวาคม 2567

สารบัญ

	หน้า
บทคัดย่อ.....	ก
Abstract.....	ข
กิตติกรรมประกาศ.....	ค
บทที่ 1 บทนำ	2
1.1 ที่มาของโครงการ	2
1.2 วัตถุประสงค์ของโครงการ.....	2
1.3 ขอบเขตของโครงการ	2
1.4 เครื่องมือที่ใช้ในการพัฒนา	2
1.6 แผนการดำเนินงาน	4
1.7 ประโยชน์ที่คาดว่าจะได้รับ	4
บทที่ 2 หลักการและทฤษฎีที่เกี่ยวข้อง.....	5
2.1 ทฤษฎีที่เกี่ยวข้อง.....	5
2.2 ตัวอย่างโครงการที่เกี่ยวข้อง	7
วิธีการดำเนินโครงการ.....	9
3.1 เครื่องมือที่ใช้ในการปฏิบัติงาน (Hardware).....	9
3.2 อุปกรณ์ที่ใช้ในการปฏิบัติโครงการ (Software).....	12
3.3 Hardware Design	14
3.4 Software Design.....	14
3.5 การออกแบบหน้าจอส่วนติดต่อผู้ใช้งาน.....	18
ผลการดำเนินงาน.....	21
4.1 ทดสอบการใช้งานระบบ	21
สรุปผลการดำเนินงาน	27

5.1 สรุปผลการทดลอง.....	27
5.2 ประโยชน์ที่ได้รับ	27
5.3 ข้อจำกัดของโครงการ.....	28
5.4 ข้อเสนอแนะ.....	28
บรรณานุกรม.....	29
ภาคผนวก.....	30
ภาคผนวก ก โค้ด ระบบตรวจจับป้องกันฝุ่นPM2.5 และแจ้งเตือนอัตโนมัติ	30
ภาคผนวก ข การทำงาน	33

สารบัญภาพ

หน้า

ภาพที่ 2-1	โครงงานเครื่องบำบัดฝุ่น PM 2.5 ด้วยระบบไฟฟ้าสถิต	7
ภาพที่ 2-2	โครงงาน ระบบเตือนภัยน้ำท่วมและวัดปริมาณฝุ่นละออง	8
ภาพที่ 3-1	NodeMCU ESP8266 V3.....	9
ภาพที่ 3-2	Active Buzzer Module.....	10
ภาพที่ 3-3	PMS5003.....	10
ภาพที่ 3-4	LCD Display	11
ภาพที่ 3-5	NodeMcu BreadBoard.....	11
ภาพที่ 3-4	Jumper Wire.....	12
ภาพที่ 3-7	Arduino IDE	12
ภาพที่ 3-8	Line Notify.....	13
ภาพที่ 3-9	Blynk.....	13
ภาพที่ 3-10	Hardware Design (การออกแบบฮาร์ดแวร์).....	14
ภาพที่ 3-11	ขั้นตอนการทำงานของการทำงานการเก็บข้อมูลวัดค่าปริมาณฝุ่นจากเซนเซอร์วัดฝุ่นPM2.5	15
ภาพที่ 3-12	ขั้นตอนการทำงานของการทำงานการควบคุม เปิด-ปิด สัญญาณการแจ้งเตือนของ Buzzer.....	16
ภาพที่ 3-13	ขั้นตอนการทำงานของการทำงานของ LCD I2C Blacklight.....	17
ภาพที่ 3-14	แสดงค่าฝุ่นPM2.5	18
ภาพที่ 3-15	กราฟแสดงค่าฝุ่นPM2.5แบบเรียลไทม์	19
ภาพที่ 3-16	การควบคุมเปิด-ปิด ของBuzzer	20
ภาพที่ 3-17	การแจ้งเตือนผ่านLineNotify	20
ภาพที่ 4-1	ทดลองระบบของอุปกรณ์.....	22
ภาพที่ 4-2	การติดตั้งSensor PMS5003	23
ภาพที่ 4-3	การติดตั้ง Buzzer.....	24
ภาพที่ 4-4	การติดตั้ง LCD Blacklight	25
ภาพที่ 4-5	ผลลัพธ์ในการทดสอบค่าฝุ่นPM2.5	26
ภาพที่ ก-1	โค้ดฟังก์ชัน การเรียกใช้LineNotify เพื่อเรียกค่าการแจ้งเตือน	30

ภาพที่ ก-2 โค้ดแสดงการดึงค่าToken ของBlynk มาใช้ในการส่งค่าและควบคุม.....	30
ภาพที่ ก-3 โค้ดการควบคุม เปิด-ปิด ของBuzzer.....	31
ภาพที่ ก-4 โค้ดแสดงการแจ้งเตือนเมื่อค่าฝุ่นมีค่าเกินกว่ากำหนด จะส่งข้อความว่า”ฝุ่นเกินค่าแล้ว” Buzzer จะส่งเสียงเตือน และเมื่อค่าฝุ่นปกติจะส่งการแจ้งเตือนว่า “ปลอดภัย”	31
ภาพที่ ก-5 โค้ดสั่งการตั้งค่าWifi ในเข้ากับตัว NodeMCU	32
ภาพที่ ก-6 โค้ดแสดงการส่งค่าของPMS5003 Sensor ส่งไปยังGauge วัดค่าฝุ่นในBlynk.....	32
ภาพที่ ข-1 การติดตั้ง NodeMCU ESP8266 ลงบนอุปกรณ์.....	33
ภาพที่ ข-2 การติดตั้งPMS5003 Sensor ลงบนอุปกรณ์	33
ภาพที่ ข-3 ติดตั้งBuzzer ลงในอุปกรณ์	34
ภาพที่ ข-4 ติดตั้ง LCD Backlight ลงบนอุปกรณ์	34
ภาพที่ ข-5 ติดตั้งเสิร์จสมบรณ์พร้อมใช้งาน.....	35
ภาพที่ ข-6 ผลลัพธ์Blynkในควบคุมอุปกรณ์ระบบตรวจจับป้องกันฝุ่นPM2.5 และแจ้งเตือนอัตโนมัติ ...	35
ภาพที่ ข-7 ผลลัพธ์ในการแจ้งเตือนค่าฝุ่นPM2.5 ที่เกินกว่ากำหนด และปลอดภัย	36

สารบัญตาราง

หน้า

ตารางที่ 1-1 แผนการดำเนินงาน.....	4
-----------------------------------	---

บทที่ 1

บทนำ

การจัดทำโครงการการตรวจจับป้องกันฝุ่นpm2.5 และแจ้งเตือนอัตโนมัติ บทนี้นำส่วนนี้จะแบ่งบอกถึงโครงการ โดยการจัดทำโครงการจะต้องประกอบไปด้วยที่มาและความสำคัญของโครงการ วัตถุประสงค์ ขอบเขตของโครงการ เครื่องมือที่ใช้ในการพัฒนา ขั้นตอนการดำเนินงาน แผนการดำเนินงาน และประโยชน์ที่คาดว่าจะได้รับ โดยผู้จัดทำได้ระบุนรายละเอียดต่างๆ ดังต่อไปนี้

1.1 ที่มาของโครงการ

ปัจจุบัน ปัญหามลพิษทางอากาศจากฝุ่นละอองขนาดเล็ก PM2.5 เป็นประเด็นที่ได้รับความสนใจและกังวลอย่างมาก เนื่องจากฝุ่น PM2.5 สามารถส่งผลกระทบต่อสุขภาพของประชาชนได้อย่างรุนแรง สาเหตุของการเกิดฝุ่นละอองดังกล่าวมาจากหลายปัจจัย เช่น การปล่อยไอเสียจากรถยนต์ การเผาป่าและขยะ รวมถึงกิจกรรมทางอุตสาหกรรมและการก่อสร้าง ฝุ่นละอองขนาดเล็กนี้สามารถเข้าสู่ระบบทางเดินหายใจและก่อให้เกิดโรคที่รุนแรง เช่น โรคทางเดินหายใจ โรคหัวใจ และมะเร็ง

เพื่อจัดการกับปัญหาดังกล่าว โครงการนี้มีวัตถุประสงค์ในการพัฒนาอุปกรณ์ตรวจจับและป้องกันฝุ่น PM2.5 ภายในบ้าน โดยใช้เทคโนโลยี Internet of Things (IoT) ซึ่งจะช่วยให้สามารถตรวจวัดปริมาณฝุ่นละอองในอากาศได้อย่างแม่นยำ และสามารถส่งข้อมูลผลการตรวจวัดไปยังอุปกรณ์ต่าง ๆ ของผู้ใช้ เช่น สมาร์ทโฟน ผ่านแอปพลิเคชัน เพื่อให้ผู้ใช้สามารถดำเนินการป้องกันและควบคุมปริมาณฝุ่นได้อย่างมีประสิทธิภาพ

1.2 วัตถุประสงค์ของโครงการ

- 1.2.1 เพื่อพัฒนาอุปกรณ์ IoT ที่สามารถตรวจจับและป้องกันฝุ่น PM2.5 ภายในบ้าน
- 1.2.2 เพื่อวัดและรายงานปริมาณฝุ่น PM2.5 ในพื้นที่ภายในบ้าน
- 1.2.3 เพื่อศึกษาการทำงานของอุปกรณ์ IoT ในการจัดการปัญหาฝุ่นละออง
- 1.2.4 เพื่อแจ้งเตือนผู้ใช้งานผ่านแอปพลิเคชัน LINE Notify เมื่อเกิดเหตุการณ์ผิดปกติ
- 1.2.5 เพื่อให้ผู้ใช้งานสามารถตรวจสอบและควบคุมระบบผ่านแอปพลิเคชัน Blynk

1.3 ขอบเขตของโครงการ

- 1.3.1 ใช้ NodeMCU ESP8266 เป็นไมโครคอนโทรลเลอร์หลักในการเชื่อมต่อและควบคุมระบบ
- 1.3.2 ใช้เซ็นเซอร์ PM5003 สำหรับตรวจจับฝุ่นและป้องกันฝุ่นpm2.5โดยเฉพาะ
- 1.3.3 ใช้จอ LCD Backlight Screen สำหรับอ่านค่าฝุ่นที่ทำการตรวจจับ
- 1.3.4 ใช้ LED สำหรับการแจ้งเตือน
- 1.3.5 ใช้ Line Notify และ Blynk Application สำหรับแจ้งเตือนและควบคุมระบบ
- 1.3.6 ระบบทำงานโดยมีการเชื่อมต่ออินเทอร์เน็ตเพื่อการแจ้งเตือนผ่านแอปพลิเคชัน
- 1.3.7 การทดสอบการทำงานของอุปกรณ์และควบคุมฝุ่นภายในบ้านจำลอง

1.4 เครื่องมือที่ใช้ในการพัฒนา

ในโครงการการตรวจจับป้องกันฝุ่น PM2.5และแจ้งเตือนอัตโนมัติ การเลือกเครื่องมือที่เหมาะสมทั้งในด้านซอฟต์แวร์ และฮาร์ดแวร์เป็นสิ่งสำคัญเพื่อให้การพัฒนา และการทำงานของระบบมีประสิทธิภาพดังต่อไปนี้

- 1.4.1 ด้านฮาร์ดแวร์
 - 1.4.1.1 NodeMCU ESP8266
 - 1.4.1.2 NodeMCUBase
 - 1.4.1.2 PM5003 Sensor
 - 1.4.1.3 LCD Backlight Screen
 - 1.4.1.4 Buzzer
 - 1.4.1.5 Jumper
- 1.4.2 ด้านซอฟต์แวร์
 - 1.4.2.1 Arduino IDE ซอฟต์แวร์สำหรับการเขียนโปรแกรมไปยัง NodeMCU ESP8266
 - 1.4.2.2 Blynk Application แอปพลิเคชันสำหรับการควบคุมระบบผ่านสมาร์ทโฟน
 - 1.4.2.3 Line Notify แจ้งเตือนผ่านแอปพลิเคชัน LINE

1.5 ขั้นตอนการดำเนินโครงการ

ผู้จัดทำโครงการได้พิจารณาความเหมาะสมและวิเคราะห์ข้อมูลอย่างละเอียดเพื่อประเมินความเป็นไปได้ของโครงการ จากผลการวิเคราะห์ข้อมูลจึงได้กำหนดขั้นตอนการดำเนินงานไว้ดังนี้

1.5.1 วิเคราะห์และการออกแบบ

1.5.2.1 ศึกษาอุปกรณ์ที่ใช้

1.5.2.2 ออกแบบการทำงานอุปกรณ์

1.5.2 การพัฒนาระบบ

1.5.2.1 การพัฒนาโปรแกรม

1.5.2.2 ทำการทดสอบระบบ

1.5.2.3 ติดตามดูการเคลื่อนไหวของอุปกรณ์ที่พัฒนาอยู่

1.5.3 แก้ไขปรับปรุงระบบ

1.5.3.1 แก้ไขปัญหาที่เกิดขึ้นให้สมบูรณ์ตามระยะเวลาการปฏิบัติงาน

1.5.4 จัดทำเอกสาร

1.5.4.1 จัดทำเล่มรายงานโครงการ

1.5.4.2 สรุปผลการทำงาน

1.5.5 นำเสนอระบบ

1.6 แผนการดำเนินงาน

การจัดทำโครงการระบบตรวจจับป้องกันฝุ่นPM2.5 และแจ้งเตือนอัตโนมัติ ในภาคเรียนที่ 2 ประจำปีการศึกษา 2567 ระยะเวลา นับตั้งแต่เดือน สิงหาคม พ.ศ. 2567 ถึงเดือน ธันวาคม พ.ศ. 2567 เป็นระยะเวลา 26 สัปดาห์ ได้ดำเนินการตามแผนงาน ดังตารางที่ 1-1

ตารางที่ 1-1 แผนการดำเนินงาน

ขั้นตอนการดำเนินงาน	ระยะเวลาในการดำเนินงาน				
	ปี พ.ศ. 2567				
	ส.ค.	ก.ย.	ต.ค.	พ.ย.	ธ.ค.
1. วิเคราะห์และออกแบบระบบ					
2. พัฒนาระบบ					
3. แก้ไขปรับปรุงระบบ					
4. จัดทำเอกสาร					
5. นำเสนอโครงการ					

1.7 ประโยชน์ที่คาดว่าจะได้รับ

- 1.7.1 พัฒนาอุปกรณ์ IOT ที่สามารถตรวจจับและจัดการปัญหาฝุ่นภายในบ้าน
- 1.7.2 ผู้ใช้งานสามารถตรวจสอบสถานะและควบคุมระบบได้ผ่านแอปพลิเคชันมือถืออย่างสะดวก
- 1.7.3 การแจ้งเตือนผ่าน Line Notify ทำให้ผู้ใช้งานได้รับข้อมูลที่สำคัญได้ตลอดเวลา

บทที่ 2

หลักการและทฤษฎีที่เกี่ยวข้อง

ในบทนี้จะกล่าวถึงหลักการ ทฤษฎี และเทคโนโลยีที่เกี่ยวข้องกับโครงงาน ระบบควบคุมควันและแจ้งเตือนด้วย IoT เพื่อให้เข้าใจถึงความสำคัญและกระบวนการทำงานของระบบที่ถูกพัฒนาขึ้น รวมถึงการอ้างอิงข้อมูลจากแหล่งความรู้ที่เกี่ยวข้อง

2.1 ทฤษฎีที่เกี่ยวข้อง

2.1.1 การพัฒนาระบบตรวจจับป้องกันฝุ่นPM2.5 และแจ้งเตือนอัตโนมัติ

แนวคิดใน การพัฒนาระบบตรวจจับป้องกันฝุ่นPM2.5 และแจ้งเตือนอัตโนมัติ สามารถแบ่งออกเป็นกระบวนการต่าง ๆ เพื่อให้ได้ระบบที่แม่นยำ เชื่อถือได้ และสามารถแจ้งเตือนได้แบบเรียลไทม์ได้จริง

1. การวิเคราะห์ความต้องการ ระบบตรวจจับป้องกันฝุ่นPM2.5 และแจ้งเตือนอัตโนมัติ ถูกออกแบบเพื่อตรวจจับฝุ่นPM1.0 2.5 และ10 เกือไนที่เป็นอันตรายต่อความปลอดภัย เช่น ฝุ่นมีค่าสูงเกินที่กำหนด ระบบต้องสามารถควบคุมการทำงานของพัดลมระบายอากาศในตัวของPMS5003 แจ้งเตือนผ่าน LINE Notify และทำงานได้แบบเรียลไทม์ นอกจากนี้ ยังต้องวิเคราะห์ปัจจัยแวดล้อมที่ส่งผลต่อการตรวจจับ รวมถึงกำหนดขอบเขตการใช้งาน เช่น การตั้งค่ามาตรฐานสำหรับการแจ้งเตือน

2. การออกแบบระบบ การออกแบบฮาร์ดแวร์เริ่มจากการเลือกเซ็นเซอร์ เช่น PMS5003 สำหรับตรวจจับฝุ่นPM2.5และระบายอากาศ, LCD Blacklight สำหรับแสดงค่าของเซ็นเซอร์PMS5003 และ NodeMCU ESP8266 สำหรับควบคุมและสื่อสารผ่านอินเทอร์เน็ต ส่วนการออกแบบซอฟต์แวร์ใช้ Arduino IDE ในการพัฒนาโค้ดสำหรับการอ่านค่าเซ็นเซอร์ การตัดสินใจควบคุมอุปกรณ์ และการส่งคำสั่งแจ้งเตือนผ่าน LINE Notify รวมถึงการออกแบบอินเทอร์เฟซที่ใช้งานง่าย เช่น หน้าจอแสดงผลข้อมูลสถานะ

3. การพัฒนาฮาร์ดแวร์ ฮาร์ดแวร์ถูกติดตั้งโดยเชื่อมต่อเซ็นเซอร์ เช่น PMS5003 กับ NodeMCU ESP8266 เพื่อรับค่าข้อมูลจากเซ็นเซอร์ ระบบควบคุมค่าฝุ่น PMS5003 และ Active Buzzer และมีจอ LCD ที่แสดงสถานะระบบ อุปกรณ์ทั้งหมดได้รับการติดตั้งในตำแหน่งที่เหมาะสมเพื่อให้ตรวจจับค่าที่ต้องการได้แม่นยำ

4. การพัฒนาซอฟต์แวร์ ระบบซอฟต์แวร์พัฒนาโดยเขียนโค้ดเพื่ออ่านค่าจากเซ็นเซอร์ต่าง ๆ และตัดสินใจควบคุมอุปกรณ์ เช่น สัญญาณ Buzzer ให้แจ้งเตือนเมื่อค่าฝุ่นสูงเกินค่ากำหนด นอกจากนี้ ยังพัฒนาระบบแจ้งเตือนผ่าน LINE Notify เพื่อส่งข้อมูลแบบเรียลไทม์ให้ผู้ใช้งาน พร้อมทั้งพัฒนาหน้าจอควบคุมที่แสดงข้อมูล เช่น ระดับค่าเฉลี่ยของค่าฝุ่นในแต่ละวัน

5. การทดสอบระบบ ระบบได้รับการทดสอบในหลายด้าน เช่น การตรวจสอบความแม่นยำของเซ็นเซอร์ การควบคุม Active Buzzer ให้ทำงานเมื่อเกิดเงื่อนไขฉุกเฉิน รวมถึงการตรวจสอบความถูกต้องของการแจ้งเตือนผ่าน LINE Notify เพื่อให้มั่นใจว่าระบบสามารถทำงานได้ตรงตามความต้องการ

6. การปรับปรุงและเพิ่มประสิทธิภาพ จากผลการทดสอบ ได้ปรับปรุงระบบ เช่น การตั้งค่าค่ามาตรฐานใหม่สำหรับเซ็นเซอร์ การปรับปรุงการทำงานของพัดลมให้เหมาะสม และเพิ่มฟีเจอร์ใหม่ เช่น การเก็บข้อมูลย้อนหลังในรูปแบบกราฟเพื่อช่วยให้ผู้ใช้งานวิเคราะห์แนวโน้มของข้อมูล นอกจากนี้ ยังปรับปรุงอินเทอร์เฟซให้ใช้งานง่ายและตอบสนองต่อความต้องการของผู้ใช้งานมากขึ้น

2.1.2 ภาษาซี (C Programming Language)

ภาษาซี (C Programming Language) เป็นภาษาโปรแกรมที่ได้รับความนิยมอย่างแพร่หลายเนื่องจากมีความเรียบง่ายและมีประสิทธิภาพสูง ทั้งยังมีอิทธิพลต่อการพัฒนาภาษาโปรแกรมอื่น ๆ อีกมากมาย เหมาะสำหรับการพัฒนาซอฟต์แวร์และระบบที่ต้องการความเร็วและความเสถียรในการทำงาน ภาษาซีมีจุดเด่นที่สำคัญ เช่น การเขียนโค้ดในรูปแบบโครงสร้าง การใช้งานฟังก์ชัน การจัดการข้อมูลพื้นฐาน ความเร็วในการประมวลผล และพอร์ตาบิลิตี้ (Portability)

ภาษาซียังมีชุดไลบรารีที่ช่วยเพิ่มความสะดวกในการพัฒนาซอฟต์แวร์ และมีการใช้งานอย่างแพร่หลายในสถาบันการศึกษา รวมถึงอุตสาหกรรมซอฟต์แวร์ แนวคิดและหลักการสำคัญของการเขียนโปรแกรมด้วยภาษาซี มีดังนี้:

1. การแบ่งโค้ดเป็นฟังก์ชัน การแยกโค้ดออกเป็นฟังก์ชันที่มีหน้าที่เฉพาะเจาะจง ทำให้โค้ดมีโครงสร้างที่ชัดเจน ง่ายต่อการจัดการ และสามารถนำฟังก์ชันกลับมาใช้งานใหม่ได้

2. การใช้งานตัวแปร ประกาศและกำหนดตัวแปรให้ชัดเจน พร้อมระบุประเภทของตัวแปรอย่างถูกต้องเพื่อช่วยให้ระบบจัดการหน่วยความจำได้อย่างมีประสิทธิภาพ

3. การใช้การดำเนินการ ใช้การดำเนินการทางคณิตศาสตร์และตรรกะอย่างเหมาะสม เพื่อช่วยในการประมวลผลข้อมูลและตรวจสอบเงื่อนไขในโค้ด

4. การใช้โครงสร้างควบคุม ใช้โครงสร้างควบคุม เช่น if, while, และ for เพื่อควบคุมลำดับการทำงานของโปรแกรมตามเงื่อนไขที่กำหนด

5. การจัดการกับอาร์เรย์ อาร์เรย์เป็นโครงสร้างข้อมูลที่สำคัญในภาษาซี ช่วยในการจัดเก็บข้อมูลหลายรายการในตัวแปรเดียว และเพิ่มความสะดวกในการจัดการข้อมูล

2.2 ตัวอย่างโครงการที่เกี่ยวข้อง

2.2.1 เครื่องบำบัดฝุ่น PM 2.5 ด้วยระบบไฟฟ้าสถิต ทำงานแบบอัตโนมัติ

เนื่องจากในปัจจุบันได้มีมลพิษฝุ่น PM 2.5 เกิดขึ้นในกรุงเทพฯและปริมณฑล ทำให้ร่างกายของผู้ที่ได้รับ ฝุ่น PM2.5 อาจจะไม่ส่งผลกระทบให้เห็นในช่วงแรกๆ แต่หากได้รับติดต่อกันเป็นเวลานาน หรือสะสมในร่างกาย สุดท้ายก็จะก่อให้เกิดอาการผิดปกติของร่างกายในภายหลัง ดังนั้นทางคณะผู้จัดทำ จึงได้จัดทำเครื่องบำบัดฝุ่น PM 2.5 ด้วยระบบไฟฟ้าสถิต ทำงานแบบอัตโนมัติ อัจฉริยะเพิ่มจะช่วยวัดปริมาณค่าฝุ่นและช่วยทำให้อากาศดีขึ้นโครงการเครื่องบำบัดฝุ่น PM 2.5 ด้วยระบบไฟฟ้าสถิตภาพที่ 2-1

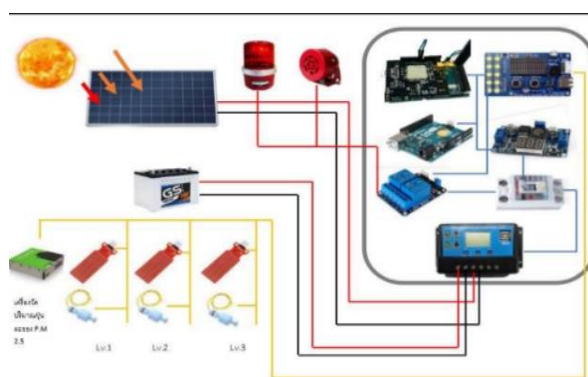


ภาพที่2-1 โครงการเครื่องบำบัดฝุ่น PM 2.5 ด้วยระบบไฟฟ้าสถิต

จากภาพที่ 2-1 เครื่องบำบัดฝุ่น PM 2.5 ด้วยระบบไฟฟ้าสถิต ทำงานแบบอัตโนมัติ ระบบตรวจจับ PM2.5 และควบคุมเครื่องบำบัดฝุ่น เขียนโดยArduinoที่รับข้อมูลมาจาก Sensor PMS3003 ส่งค่าให้กับ Relay เพื่อสั่ง ให้ควบคุมการเปิด-ปิด ชุดวงจรไฟแรงสูง(high voltage circuit) เครื่องบำบัดฝุ่น

2.2.2 โครงการ ระบบเตือนภัยน้ำท่วมและวัดปริมาณฝุ่นละออง

เนื่องด้วยในปัจจุบันเทคโนโลยีคอมพิวเตอร์และเทคโนโลยีของระบบ Smart Phone ได้ก้าวหน้าไปอย่างรวดเร็วมี Application ต่างๆ ออกมามากมาย เพื่ออำนวยความสะดวกในการใช้ชีวิตของมนุษย์ ทำให้ สามารถควบคุมอุปกรณ์ได้อย่างสะดวกสบายง่ายและรวดเร็ว โดยผ่าน Application ซึ่งมีให้ดาวน์โหลดฟรี และแบบเขียนขึ้นเอง ทำให้สามารถเชื่อมโยงระหว่างกันและกันผ่านอินเทอร์เน็ต เครือข่ายในลักษณะนี้ เรียกว่า The Internet of Things เรียกอย่อ ๆ ว่า IOTซึ่งแสดงให้เห็นในภาพที่ 2-2



ภาพที่ 2-2 โครงการ ระบบเตือนภัยน้ำท่วมและวัดปริมาณฝุ่นละออง

จากภาพที่ 2-2 โครงการ ระบบเตือนภัยน้ำท่วมและวัดปริมาณฝุ่นละออง ระบบเตือนภัยน้ำท่วมและเครื่องวัดปริมาณฝุ่นละออง PM 2.5 หลังจากได้ทำการทดลองทำโครงการได้ พบปัญหาในเรื่องของเซนเซอร์วัดระดับน้ำเนื่องจากเมื่อเซนเซอร์โดนน้ำก็จะทำงานทันทีหากสมมุติว่าน้ำที่โดน ไม่ใช่มาจากแม่น้ำจริงแต่อาจเป็นน้ำฝนเป็นต้น อาจทำให้แจ้งระดับน้ำผิดพลาด ผู้จัดทำจึงได้ปรึกษากับครูที่ปรึกษาและทางวิทยากรที่มาให้ความรู้ ผลสรุปได้มาว่า ได้เพิ่มสวิตช์กลอยเพื่อเป็นการยืนยันสถานะของ ระดับน้ำจริง ทำให้โครงการสมบูรณ์มากกว่าเดิม สรุปผลการทำโครงการ โครงการระบบเตือนภัยน้ำท่วมและเครื่องวัดปริมาณฝุ่นละออง สามารถส่งค่าระดับน้ำและปริมาณฝุ่นละออง PM 2.5 ผ่านทาง Application Line ให้กับชาวบ้านที่อยู่บริเวณพื้นที่เสี่ยงภัยและผู้ควบคุมดูแลให้ทราบ

บทที่ 3

วิธีการดำเนินโครงการ

บทนี้จะกล่าวถึงรายละเอียดเครื่องมือและเทคโนโลยีที่ใช้ในการปฏิบัติโครงการระบบตรวจจับป้องกันฝุ่นPM2.5 และแจ้งเตือนอัตโนมัตินั้นประกอบไปด้วยเครื่องมือที่ใช้ในการพัฒนา และภาษาที่ใช้ในการพัฒนา โดยมีรายละเอียดดังนี้

3.1 เครื่องมือที่ใช้ในการปฏิบัติงาน (Hardware)

3.1.1 NodeMCU ESP8266 V3

บอร์ดที่ใช้ในการเชื่อมวงจรต่างๆเข้าด้วยกันแสดงดังภาพที่ 3-1



ภาพที่3-1 NodeMCU ESP8266 V3

จากภาพที่ 3-1 Node MCU V3 คือ บอร์ดไมโครคอนโทรลเลอร์ที่มีลักษณะการทำงานตามคำสั่งภาษา C คล้าย Arduino แต่มีลักษณะพิเศษกว่าตรงที่สามารถเชื่อมต่อกับ Wi-Fi ได้ ด้านการควบคุมการทำงาน สามารถใช้โปรแกรม Arduino IDE

3.1.2 Active Buzzer Modul

อุปกรณ์อิเล็กทรอนิกส์ที่ใช้ในการสร้างเสียงเตือนหรือสัญญาณเสียงแสดงดังภาพที่ 3-2



ภาพที่ 3-2 Active Buzzer Module

จากภาพที่ 3-2 Active Buzzer Module ลำโพงแบบแม่เหล็กหรือแบบ piezo ที่มีวงจรถูกกำหนดความถี่ อยู่ภายในตัว และใช้ไฟเลี้ยง 3.3-5V

3.1.3 PMS5003

เซ็นเซอร์สำหรับตรวจจับฝุ่น PM2.5 และป้องกันฝุ่นแสดงดังภาพที่ 3-3



ภาพที่ 3-3 PMS5003

จากภาพที่ 3-3 PMS5003 เป็นเซ็นเซอร์ตรวจจับฝุ่นละอองขนาดเล็กในอากาศ ซึ่งสามารถวัดค่าฝุ่นละอองที่มีขนาดต่างๆ เช่น PM1.5 PM2.5 และ PM10 โดยมักใช้ในระบบตรวจจับคุณภาพอากาศ เช่น ระบบ IoT สำหรับการตรวจสอบคุณภาพอากาศภายในบ้าน อาคาร หรือสถานที่ต่างๆ เพื่อติดตามและแจ้งเตือนสภาพอากาศที่มีมลพิษจากฝุ่นละอองในอากาศ

3.1.4 LCD Display

พัลลัมไฟฟ้ากระแสตรงที่ทำงานด้วยแรงดันไฟฟ้า 5 โวลต์แสดงดังภาพที่ 3-4

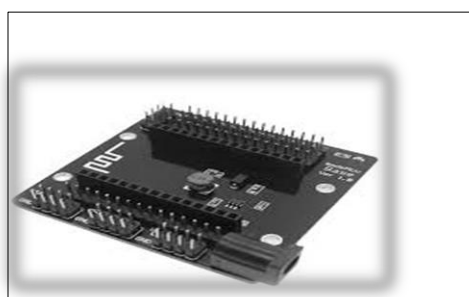


ภาพที่ 3-4 LCD Display

จากภาพที่ 3-4 LCD Display เป็นเทคโนโลยีการแสดงผลที่ใช้ในการแสดงข้อมูลหรือภาพต่างๆ โดยใช้ผลของผลึกเหลว ในการควบคุมการผ่านของแสง เพื่อให้แสดงเป็นข้อความหรือภาพที่ผู้ใช้งานสามารถมองเห็นได้ โดยทั่วไปแล้ว LCD จะใช้ในอุปกรณ์ต่างๆ รวมถึงการแสดงผลในระบบ IoT และการควบคุมเครื่องมือในงานวิจัยและการพัฒนา

3.1.5 NodeMcu Breadboard

เซ็นเซอร์วัดอุณหภูมิและความชื้นที่มีความแม่นยำสูงแสดงดังภาพที่ 3-5



ภาพที่ 3-5 NodeMcu BreadBoard

จากภาพที่ 3-5 NodeMcu BreadBoard เป็นโมดูลไมโครคอนโทรลเลอร์ที่ใช้ชิป ESP8266 ซึ่งเป็นชิป Wi-Fi ที่สามารถเชื่อมต่อกับอินเทอร์เน็ตได้ ทำให้ NodeMCU เหมาะสำหรับการพัฒนาและทดลองโปรเจกต์ต่างๆ ที่เกี่ยวข้องกับ IoT หรือการเชื่อมต่ออุปกรณ์ต่างๆ กับเครือข่ายอินเทอร์เน็ต

3.1.6 Jumper Wire

สายไฟที่ใช้ในการเชื่อมต่ออุปกรณ์ต่าง ๆ เข้าด้วยกันแสดงดังภาพที่ 3-7



ภาพที่ 3-4 Jumper Wire

จากภาพที่ 3-6 Jumper Wire คือสายไฟที่ใช้ในการเชื่อมต่ออุปกรณ์อิเล็กทรอนิกส์ต่างๆ บนบอร์ดวงจร เช่น Arduino หรือ Breadboard โดยมีปลายขั้วที่เป็นหัวเสียบแบบตัวผู้ (Male) หรือ ตัวเมีย (Female) เพื่อให้สะดวกในการเชื่อมต่อระหว่างอุปกรณ์

3.2 อุปกรณ์ที่ใช้ในการปฏิบัติโครงการ (Software)

3.2.1 Arduino IDE โปรแกรมที่ใช้ในการควบคุมการทำงานของบอร์ดแสดงดังภาพ 3-7

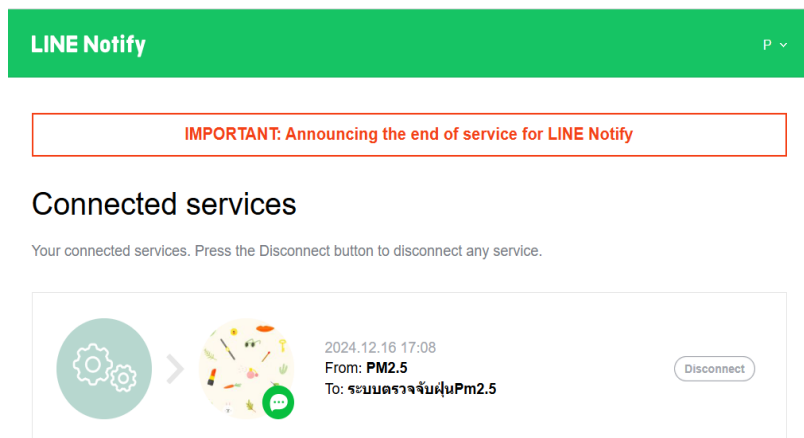


ภาพที่ 3-7 Arduino IDE

จากภาพที่ 3-7 Arduino IDE คือซอฟต์แวร์ที่ใช้ในการเขียนโปรแกรมและอัปโหลดโค้ดไปยังบอร์ด Arduino โดย Arduino IDE เป็นเครื่องมือที่ใช้งานง่ายและเป็นที่ยอมรับในหมู่ผู้เริ่มต้นและนักพัฒนา

3.2.2 LINE Notify

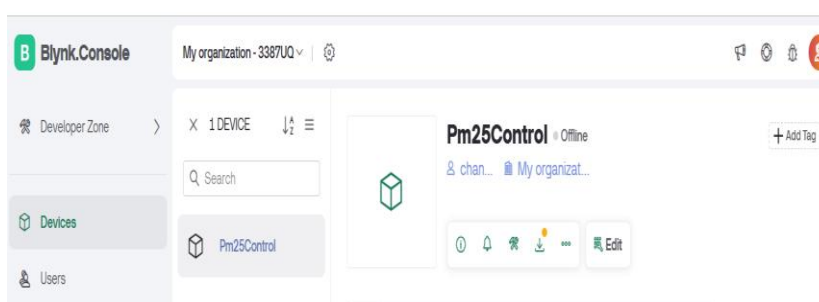
แพลตฟอร์มที่ใช้แสดงการแจ้งเตือนต่างๆแสดงดังภาพ 3-8



ภาพที่ 3-8 Line Notify

จากภาพที่ 3-8 Line Notify คือบริการที่ให้ผู้ใช้งานสามารถส่งข้อความแจ้งเตือน ไปยังบัญชี LINE ของตนเองหรือกลุ่ม LINE โดยใช้ Tokenซึ่งทำให้สามารถเชื่อมต่อแอปพลิเคชันหรือระบบต่างๆ กับ LINE ได้ง่ายและสะดวก บริการนี้มักถูกนำมาใช้ในโปรเจกต์หรือแอปพลิเคชันที่ต้องการส่งข้อความหรือการแจ้งเตือนแบบอัตโนมัติให้กับผู้ใช้ ผ่าน LINE

3.2.3 Blynkโปรแกรมที่ใช้ควบคุมการทำงานของอุปกรณ์แสดงดังภาพ 3-9

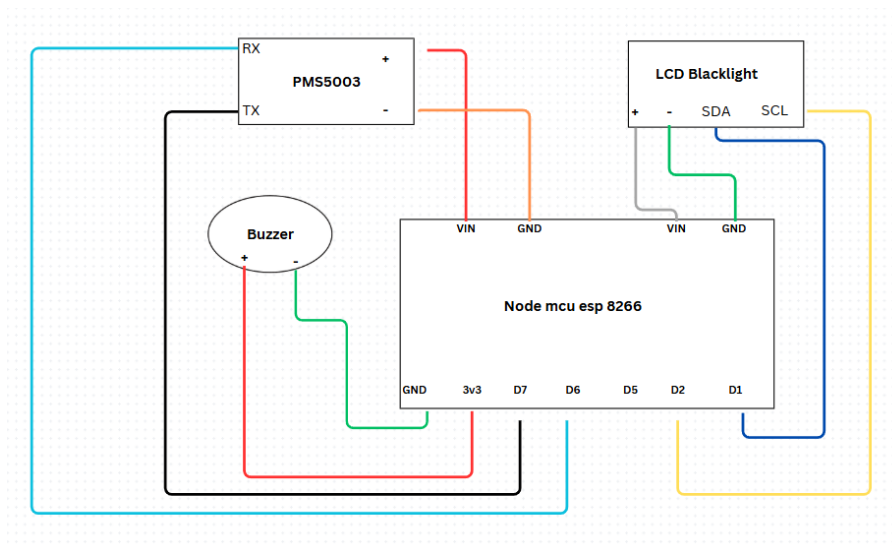


ภาพที่ 3-9 Blynk

จากภาพที่ 3-9 Blynk คือแพลตฟอร์มและบริการที่ช่วยให้ผู้ใช้สามารถเชื่อมต่ออุปกรณ์ ต่างๆ กับระบบออนไลน์หรือแอปพลิเคชันได้อย่างง่ายดาย โดยไม่จำเป็นต้องพัฒนาโค้ดหรือซอฟต์แวร์เองทั้งหมด ซึ่ง Blynk ให้บริการเชื่อมต่อและควบคุมอุปกรณ์ IoT ผ่านคลาวด์

3.3 Hardware Design

ในส่วนของ Hardware Design นั้นผู้จัดทำได้ทำการสร้างขึ้นมาเพื่อ ออกแบบและการทำงานของ ชุดควบคุมระบบตรวจจับป้องกันฝุ่นPM2.5 และแจ้งเตือนอัตโนมัติแสดงภาพที่ 3-10



ภาพที่ 3-10 Hardware Design (การออกแบบฮาร์ดแวร์)

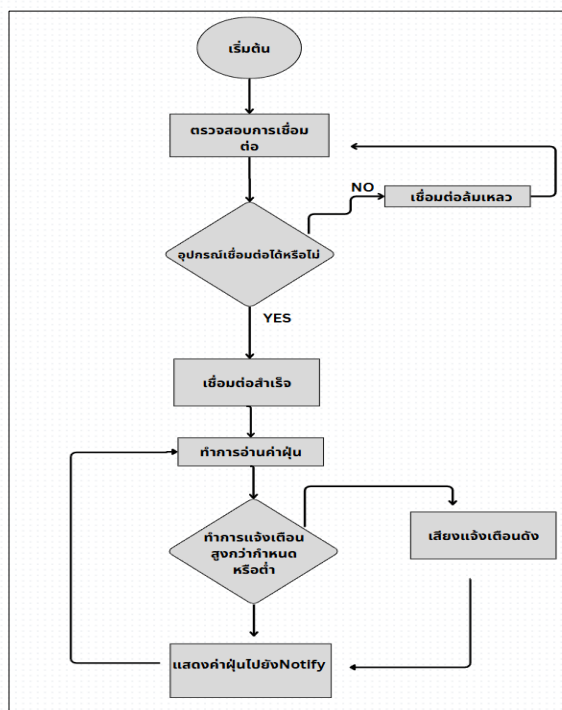
จากภาพ 3-10 แสดงการต่ออุปกรณ์ทั้งหมดเข้าที่ Node MCU ESP 8266 ดังนี้

1. NodeMCU ESP8266
2. PMS5003 ขา RX ไปที่ D6 และ TX ไปที่ D7 และตามด้วย GND (-) VIN (+)
3. LCD I2C 16x2 เชื่อม SDA ไปที่ D1 และ SCL ไปที่ D2 และตามด้วย GND (-) VIN (+)
4. Buzzer เชื่อมต่อกับขา D5 GND(-) และ 3v3 (+)

3.4 Software Design

ทางผู้จัดทำได้วิเคราะห์ขั้นตอนและกระบวนการของระบบตรวจจับป้องกันฝุ่นPM2.5 และแจ้งเตือนอัตโนมัติ โดยมีระบบดังต่อไปนี้ ระบบแจ้งเตือนฝุ่น การแสดงค่าฝุ่น การแสดงว่าค่าฝุ่นเกินแจ้งเตือนไปยัง Line Notify และมีการส่งผ่านการควบคุมการเปิด-ปิด ของ Buzzer เมื่อค่าฝุ่นเกินกำหนดและต้องการปิดเสียงแจ้งเตือนให้เงียบลง

3.4.1 ขั้นตอนการทำงานของการทำงานการตรวจวัดปริมาณฝุ่นPM2.5



ภาพที่ 3-11 ขั้นตอนการทำงานของการทำงานการเก็บข้อมูลวัดค่าปริมาณฝุ่นจากเซนเซอร์วัดฝุ่นPM2.5

จากภาพที่ 3-11 แสดงขั้นตอนการทำงานของการทำงานการเก็บค่าฝุ่นPM2.5 โดยมาจากเซนเซอร์ PMS5003 โดยจะมีลำดับขั้นตอนดังนี้

เริ่มต้น เริ่มต้นด้วยการใช้ เริ่มต้น ที่บล็อกแรกเพื่อกำหนดจุดเริ่มต้นของโปรแกรม
ตรวจสอบสถานะการเชื่อมต่อ อุปกรณ์

ตรวจสอบการเชื่อมต่ออุปกรณ์ สำเร็จถ้าการเชื่อมต่อสำเร็จ เชื่อมต่อแล้ว จะเชื่อมต่อไปยังขั้นตอนถัดไป

ไม่สำเร็จ ถ้าการเชื่อมต่อ ไม่สำเร็จ เชื่อมต่อไม่ได้ ให้วนลูปกลับไป "เชื่อมต่อ" จนกว่าจะเชื่อมต่อได้

ตรวจสอบการตั้งค่าเวลา หลังจากกำหนดเวลาแล้ว ตรวจสอบว่าเวลาถูกต้องหรือไม่ ถ้าเวลาถูกต้องให้ดำเนินการต่อไปที่การเริ่มการตรวจวัด PM2.5

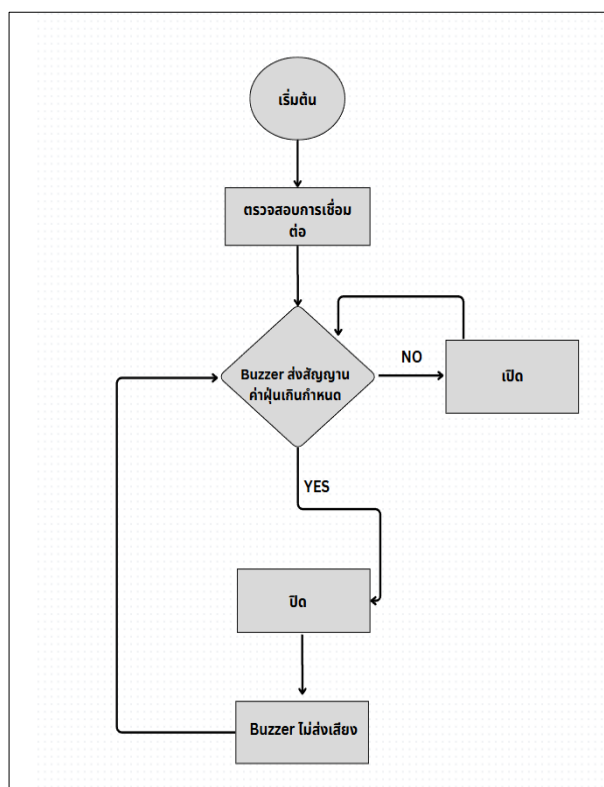
ตรวจวัดค่า PM2.5 เมื่อได้เวลาแล้ว ระบบจะเริ่มการตรวจวัดค่า PM2.5 โดยใช้เซนเซอร์ PMS5003

การแจ้งเตือน ถ้าค่าฝุ่น PM2.5 สูงกว่าค่าที่กำหนด เกินเกณฑ์ที่ตั้งไว้ ให้ส่งการแจ้งเตือนผ่าน LINE Notify และเปิดสัญญาณเสียง Buzzer

เสร็จสิ้นการตรวจวัด ถ้าค่าฝุ่นต่ำกว่าค่าที่กำหนด ระบบจะทำการหยุดการเตือนและส่งข้อความที่แสดงว่า "ปลอดภัย"

การทำงานวนลูป เมื่อทำการตรวจวัดเสร็จสิ้นแล้ว ระบบจะกลับไปตรวจวัดอีกครั้งในรอบถัดไป

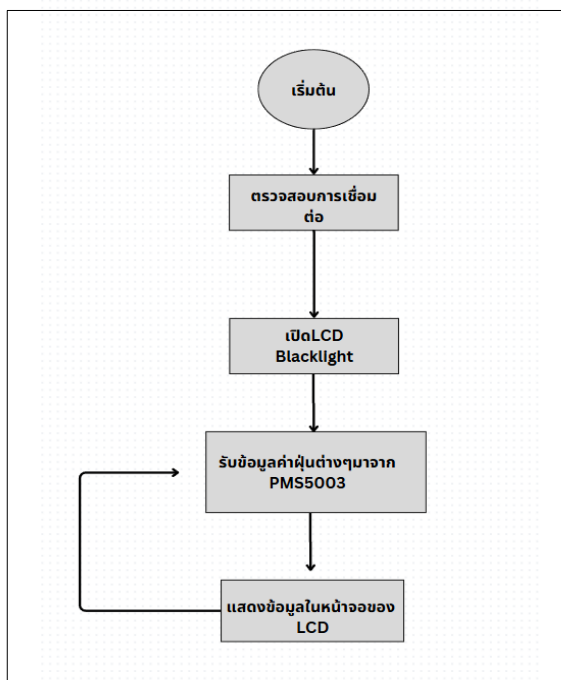
3.4.2 ขั้นตอนการทำงานของการทำงานควบคุมเปิด-ปิด สัญญาณเสียงการแจ้งเตือนของBuzzer



ภาพที่ 3-12 ขั้นตอนการทำงานของการทำงานควบคุม เปิด-ปิด สัญญาณการแจ้งเตือนของ Buzzer

จากภาพที่ 3-12 แสดงขั้นตอนการทำงานของการทำงานควบคุม เปิด-ปิด สัญญาณการแจ้งเตือนของ Buzเริ่มต้น ตรวจสอบสัญญาณจาก Blynk Buzzer Command ถ้าBuzzer Command เป็น 1ปิด Buzzer ส่งสัญญาณให้ Buzzerทำงาน)ถ้า Buzzer Command เป็น 0ปิด Buzzer หยุดการทำงานของ Buzzer

3.4.3 ขั้นตอนการทำงานของ LCD I2C Blacklight



ภาพที่ 3-13 ขั้นตอนการทำงานของ LCD I2C Blacklight

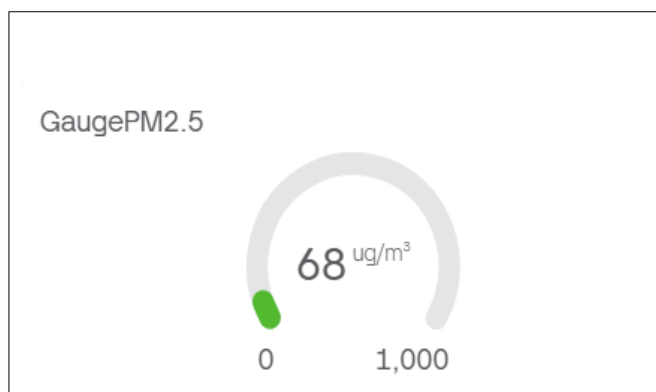
จากภาพที่ 3-13 ขั้นตอนการทำงานของ LCD I2C Blacklight เมื่อเริ่มต้นการทำงานของโปรแกรม ระบบจะเริ่มทำงานจากฟังก์ชัน `setup()` ซึ่งในขั้นตอนนี้จะเตรียมสิ่งต่าง ๆ ที่ต้องการสำหรับโปรแกรม เช่น การตั้งค่าพอร์ตการเชื่อมต่อ เมื่อข้อมูลถูกแสดงผลเสร็จสิ้นแล้ว โปรแกรมอาจดำเนินการต่อไปในขั้นตอนอื่น ๆ หรือจะรอการทำงานในระยะยาวจากเซ็นเซอร์ (เช่น การตรวจวัดค่า PM2.5 ซ้ำ) ขึ้นอยู่กับการตั้งค่าของโปรแกรมในส่วนอื่น ๆ เมื่อโปรแกรมเสร็จสิ้นการทำงานกับ LCD จะแสดงผลลัพธ์ตามที่กำหนดไว้ในโค้ด

3.5 การออกแบบหน้าจอส่วนติดต่อผู้ใช้งาน

การออกแบบหน้าจอส่วนผู้ใช้งาน จะมีการอ้างอิงจากแอปพลิเคชัน หรือเว็บไซต์การออกแบบ เนื่องจากชุดควบคุมระบบการตรวจจับป้องกันฝุ่นPM2.5 และการแจ้งเตือนอัตโนมัติ เป็นตัวควบคุมทำงานเกือบทั้งหมดการออกแบบในส่วนของผู้ใช้งานมีดังนี้

3.5.1 แสดงค่าปริมาณฝุ่น

ค่าฝุ่นPM2.5 ที่เก็บจาก PMS5003 Sensor แสดงดังภาพที่ 3-14



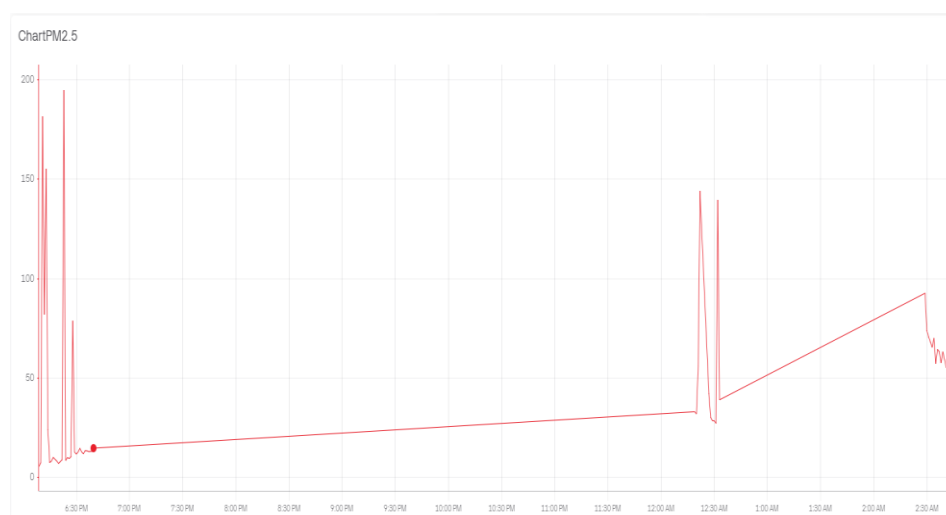
ภาพที่ 3-14 แสดงค่าฝุ่นPM2.5

จากภาพที่ 3-14 แสดงค่าฝุ่นPM2.5 ที่ถูกตรวจจับโดยเซนเซอร์ตรวจจับฝุ่นหรือ (PMS5003) ซึ่งจะมีหน้าที่วัดค่าฝุ่นทั้งหมด และส่งค่าไปยังบอร์ดเพื่อทำการประมวลค่าของฝุ่นที่เซนเซอร์นั้นตรวจจับได้ และส่งค่าไปยังBlynk ให้ผู้ใช้งานได้อ่านค่าของฝุ่นแบบเรียลไทม์ได้

การที่นำเสนอข้อมูลลงไปยังBlynk ทำให้ผู้ใช้งาน ใช้งานได้อย่างสะดวกและรู้ค่าได้ทันถ่วงทีจะได้ไม่เกิดอันตรายกับสุขภาพ เมื่อค่าฝุ่นมีปริมาณมาก ในการที่เราสามารถรู้ค่าฝุ่นได้อย่างทันถ่วงที เราก็จะแก้ไขและปลอดภัยจากฝุ่นPM2.5 ได้

3.5.2 กราฟแสดงช่วงเวลาของค่าปริมาณฝุ่นPM2.5แบบเรียลไทม์

กราฟแสดงช่วงเวลาของค่าฝุ่นแสดงดังภาพที่ 3-15



ภาพที่ 3-15 กราฟแสดงค่าฝุ่นPM2.5แบบเรียลไทม์

จากภาพที่3-15 กราฟแสดงค่าฝุ่นPM2.5แบบเรียลไทม์กราฟแสดงค่าฝุ่น PM2.5 ใน Blynk จะช่วยแสดงข้อมูลในรูปแบบกราฟที่สามารถมองเห็นการเปลี่ยนแปลงของค่า PM2.5 ในแต่ละช่วงเวลาได้ชัดเจน เช่น การเปลี่ยนแปลงในระดับชั่วโมง หรือวัน ขึ้นอยู่กับการตั้งค่าของโปรแกรม

กราฟช่วยให้สามารถกำหนดขีดจำกัดที่แสดงถึงค่าฝุ่นที่อันตราย เช่น เมื่อค่า PM2.5 เกินระดับที่กำหนด (เช่น 100 $\mu\text{g}/\text{m}^3$) จะช่วยให้ผู้ใช้เห็นภาพรวมได้ง่ายกว่าค่าฝุ่นในพื้นที่เริ่มมีความเสี่ยง

การแสดงผลกราฟนี้จะช่วยในการเตือนภัยและการรับรู้สถานการณ์การปนเปื้อนฝุ่นที่อาจเป็นอันตรายต่อสุขภาพ

ผู้ใช้สามารถติดตามข้อมูลการตรวจวัดค่าฝุ่น PM2.5 ได้แบบเรียลไทม์ ผ่านแอป Blynk โดยไม่ต้องติดตามจากอุปกรณ์ที่วัดค่าฝุ่นโดยตรง

3.5.3 การควบคุมการเปิด-ปิด ของBuzzer

การควบคุมเปิด-ปิดของBuzzerแสดงดังภาพที่ 3-16

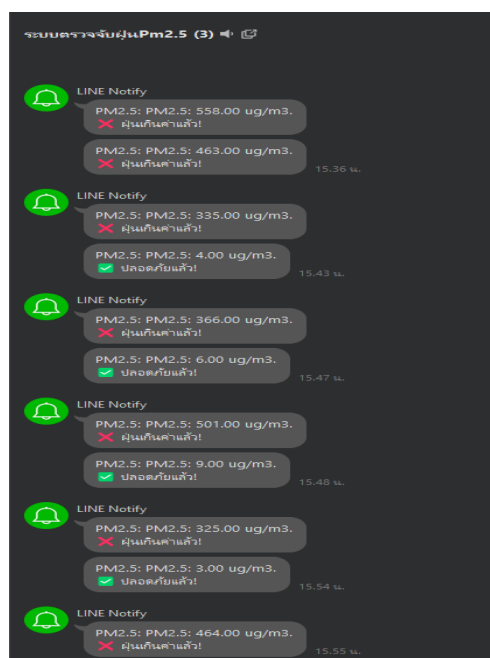


ภาพที่ 3-16 การควบคุมเปิด-ปิด ของBuzzer

จากภาพที่ 3-16 การควบคุมการเปิด-ปิดของBuzzer เป็นการให้ผู้ที่ใช้งานBlynk ได้ใช้งานเวลาที่ค่าฝุ่นเกินที่ค่ากำหนดไว้ตัวBuzzerนั้น จะทำการส่งเสียงสัญญาณเตือน เป็นเสียงที่ค่อนข้างดังเพื่อให้สัญญาณอันตรายที่เกิดกับผู้ใช้งานได้รู้

3.5.4 การแจ้งเตือนผ่านLineNotify

การแจ้งเตือนผ่านทางLineApplicationแสดงดังภาพที่ 3-17



ภาพที่ 3-17 การแจ้งเตือนผ่านLineNotify

จากภาพที่ 3-17 แสดงการแจ้งเตือนผ่านLine เป็นฟังก์ชันที่มีการแจ้งเตือนถึงค่าของฝุ่นPM2.5 ที่ส่งมาจากบอร์ดทำให้เราได้รับรู้ถึงค่าต่างๆที่เราได้กำหนดไว้ แจ้งเตือนทั้งฝุ่นเกิน และค่าฝุ่นปลอดภัยแล้ว

บทที่ 4

ผลการดำเนินงาน

จะกล่าวถึงผลการดำเนินงานของโครงการระบบตรวจจับป้องกันฝุ่นPM2.5 และแจ้งเตือนอัตโนมัติ โดยจะมีระยะการปฏิบัติงานได้มีการนำทักษะ และความรู้ต่างๆ เข้ามาประยุกต์ในระหว่างปฏิบัติงาน โดยประกอบไปด้วยผลการทดลองโครงการที่มีรายละเอียด

4.1 ทดสอบการใช้งานระบบ

4.1.1 ขั้นตอนการทดลอง

- 1 ติดตั้งอุปกรณ์ Hardware
2. ทำการเปิดโปรแกรม Arduino
3. ทดลองSensorและModuleทั้งหมดว่าใช้งานได้ไหม
4. ลองรันโค้ดเพื่อทดสอบระบบของแต่ละSensorและModule
5. ลองประกอบทุกอย่างเข้าด้วยกันและเริ่มปฏิบัติขึ้นงาน
6. ทดลองค่าฝุ่นว่าสัมพันธ์กับจอLCD ว่าส่งค่า
7. ทดลองระบบการควบคุมการตั้งเวลาเปิดปิดไฟ
- 8.ทดสอบการแจ้งเตือนเมื่อค่าฝุ่นเกินกำหนดจะส่งผ่านLine Notify
- 9.มีการควบคุมการส่งสัญญาณเตือนของBuzzer ได้
- 10.เช็คค่าฝุ่นและกราฟแสดงค่าฝุ่นแบบเรียลไทม์ได้ในBlynk ผ่านSmartPhone

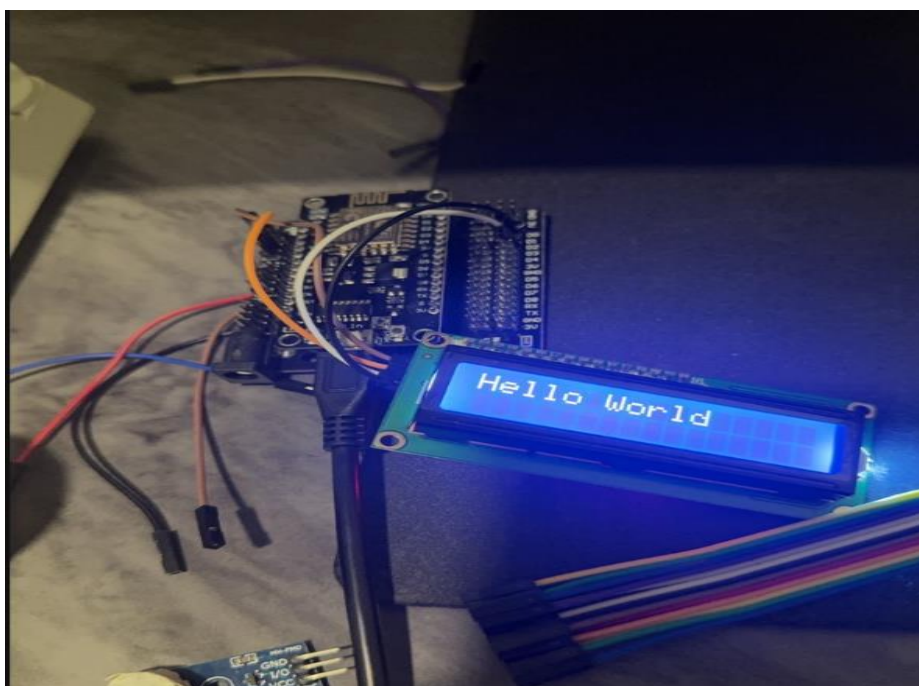
ในการทำงานของระบบตรวจจับฝุ่นป้องกันฝุ่นPM2.5 และแจ้งเตือนอัตโนมัติ กับEsp-8266 อุปกรณ์ต่างๆ

ที่จะนำมาพัฒนาระบบตรวจจับป้องกันฝุ่นPM2.5 และแจ้งเตือนอัตโนมัติ

4.1.2 ผลการทดลอง

ผลการทดลองของระบบตรวจจับฝุ่นป้องกันPM2.5 และแจ้งเตือนอัตโนมัติและการออกแบบจำลองขึ้นโดยมีรายละเอียดดังนี้

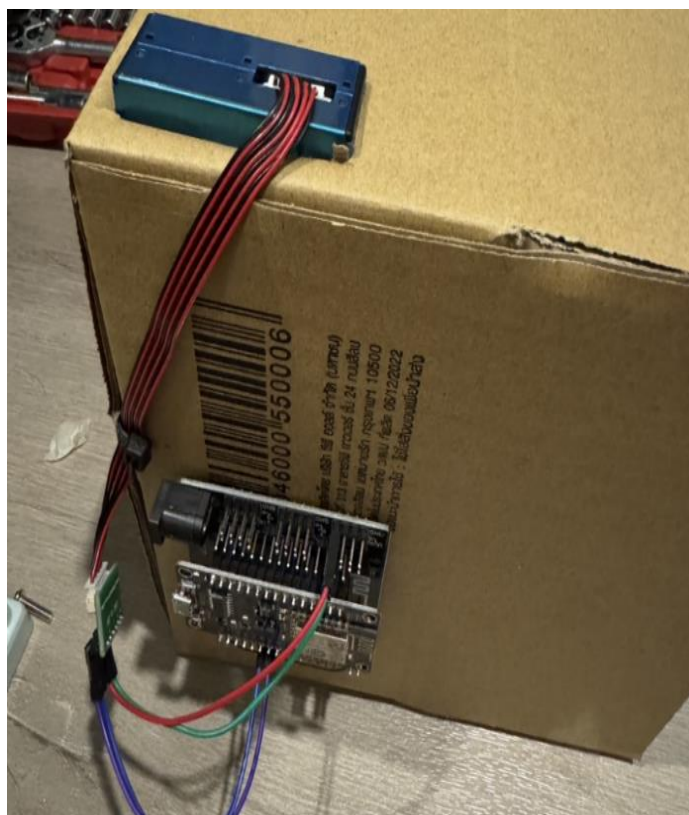
4.1.2.1 ทดลองระบบของอุปกรณ์ภาพที่ 4-1



ภาพที่ 4-1 ทดลองระบบของอุปกรณ์

จากภาพที่ 4-1 แสดงการทดลองระบบของอุปกรณ์ว่าเสียหายชำรุด ใช้ได้หรือไม่ โดยมีการนำเครื่องมือเหล่านั้นมาต่อเข้ากับBoardMCU แล้วทำการรันโค้ดTEST ดูว่าระบบใช้ได้ตามปกติหรือไม่

4.1.2.2 ติดตั้งSensor PMS5003 ดังภาพที่ 4-2



ภาพที่ 4-2 การติดตั้งSensor PMS5003

จากภาพที่ 4-2 แสดงการติดตั้งSensor ที่มีชื่อว่า PMS5003 เป็น Sensor ที่ทำหน้าที่ในการตรวจจับฝุ่นต่างๆเช่น PM1.0 PM2.5 PM10 เป็นต้น

ในโครงการของผู้จัดทำนี้ ได้สังเกตเห็นในการทดลองของค่าฝุ่น PM2.5 เพราะเป็นที่ระบาดในบ้านเมืองมากที่สุด จึงได้คิดค้นและพัฒนาระบบโดย มีSensor ที่มีชื่อว่า PMS5003

4.1.2.3 ติดตั้ง Buzzer ดังภาพที่ 4-3



ภาพที่ 4-3 การติดตั้ง Buzzer

จากภาพที่ 4-3 แสดงการติดตั้ง Buzzer โดย Buzzer นั้นเป็นModule ที่ส่งเสียงแจ้งเตือนที่ตั้งเพื่อบอกเหตุที่กำลังเกิดอยู่ โดยการทำให้โครงงานระบบตรวจจับป้องกันฝุ่นPM2.5 และแจ้งเตือนอัตโนมัติได้นำตัว Buzzer นี้เข้ามาใช้มาพัฒนาเพราะว่า

เมื่อทำการตั้งค่าต่างๆให้กับบอร์ดแล้วว่าค่าฝุ่นมีเท่าใด ถ้าหากไม่มีBuzzer ผู้ใช้งานจะไม่ทราบว่าค่าฝุ่นเกินหรือไม่ แต่เมื่อเรามี Module ตัวนี้มาในการพัฒนาระบบเลยทำให้การพัฒนาในมีคุณภาพนั้นเนื่องจากสามารถ กำหนด ค่าให้สัญญาณส่งเสียงร้องเมื่อมีค่าฝุ่นตามที่กำหนด

4.1.2.4 ติดตั้งจอ LCD BlackLight



ภาพที่ 4-4 การติดตั้ง LCD Blacklight

จากภาพที่ 4-4 แสดงให้เห็นถึงการติดตั้งและการใช้งาน จอ LCD I2C Blacklight ซึ่งถือเป็นองค์ประกอบสำคัญในระบบตรวจจับและป้องกันฝุ่น PM2.5 ที่ได้พัฒนาขึ้น จอ LCD I2C ถูกเลือกใช้งานในโครงการนี้เนื่องจากมีความสามารถในการแสดงข้อมูลได้อย่างชัดเจนและประหยัดพลังงาน โดยใช้ I2C Interface ที่ช่วยลดจำนวนขาที่ต้องเชื่อมต่อกับไมโครคอนโทรลเลอร์ ทำให้การออกแบบวงจรมีความเรียบง่ายและมีประสิทธิภาพ

4.1.2.5 ผลลัพธ์ในการทดสอบค่าฝุ่นPM2.5 ให้แจ้งเตือนในSmartphone



ภาพที่ 4-5 ผลลัพธ์ในการทดสอบค่าฝุ่นPM2.5

จากภาพที่ 4-5 แสดงผลลัพธ์ของการทดสอบระบบตรวจจับค่าฝุ่น PM2.5 และการแจ้งเตือนผ่านสมาร์ทโฟน โดยใช้ LINE Notify เป็นช่องทางในการสื่อสารข้อมูลกับผู้ใช้งาน ระบบสามารถทำการส่งข้อความแจ้งเตือนอัตโนมัติเมื่อค่าฝุ่น PM2.5 มีการเปลี่ยนแปลง โดยผลลัพธ์ในการทดสอบสามารถสรุปได้ดังนี้

แจ้งเตือนเมื่อค่าฝุ่นเกินมาตรฐาน ระบบจะส่งข้อความแจ้งเตือนพร้อมข้อมูลค่าของฝุ่นPM2.5 เมื่อระดับค่าฝุ่นเกินกำหนดเช่น 50ug/m3 ข้อความเตือนคือ PM2.5: 50 ug/m3 ก็จะบอกว่าค่าฝุ่นที่เกินกำหนด

แจ้งเตือนเมื่อค่าฝุ่นปกติ ระบบจะส่งข้อความแจ้งเตือนพร้อมกับข้อมูลค่าของฝุ่นPM2.5 น้อยเป็นปกติ จะส่งข้อความเช่น “PM2.5: 20ug/m3” ค่าฝุ่นปลอดภัยแล้ว

บทที่ 5

สรุปผลการดำเนินงาน

5.1 สรุปผลการทดลอง

ระบบตรวจจับและป้องกันฝุ่น PM2.5 ที่พัฒนาขึ้นในโครงงานนี้สามารถทำงานได้อย่างสมบูรณ์ และตรงตามเป้าหมายที่กำหนดไว้ โดยเซ็นเซอร์ PMS5003 ทำหน้าที่ตรวจจับค่าฝุ่น PM2.5 ในอากาศได้อย่างแม่นยำ และส่งข้อมูลไปแสดงผลบนจอ LCD I2C Blacklight แบบเรียลไทม์ ซึ่งช่วยให้ผู้ใช้งานสามารถตรวจสอบค่าฝุ่นปัจจุบันได้ทันที นอกจากนี้ ระบบยังมีฟังก์ชันแจ้งเตือนผ่าน LINE Notify ที่ทำงานได้อย่างมีประสิทธิภาพ สามารถแจ้งเตือนเมื่อค่าฝุ่นเกินมาตรฐานและเมื่อค่าฝุ่นกลับสู่ระดับปลอดภัยได้อย่างทันท่วงที

การพัฒนาระบบยังรวมถึงการควบคุมการแจ้งเตือนเสียงผ่าน Buzzer ซึ่งผู้ใช้งานสามารถเปิดหรือปิดเสียงได้ตามความต้องการผ่านแอปพลิเคชัน Blynk ระบบยังเชื่อมต่อกับ Blynk เพื่อแสดงค่าฝุ่น PM2.5 ในรูปแบบกราฟแบบเรียลไทม์ ช่วยให้ผู้ใช้งานสามารถติดตามและวิเคราะห์แนวโน้มค่าฝุ่นในระยะยาวได้

ผลการทดลองแสดงให้เห็นว่าระบบมีความเสถียรและสามารถทำงานได้ต่อเนื่องโดยไม่มีข้อผิดพลาด ความสามารถของระบบนี้ช่วยลดความเสี่ยงด้านสุขภาพของผู้ใช้งาน โดยให้ข้อมูลที่ชัดเจนและการแจ้งเตือนที่ทันสมัย การพัฒนานี้เป็นตัวอย่างที่ดีของการนำเทคโนโลยี IOT มาใช้ในชีวิตประจำวัน เพื่อแก้ไขปัญหาสิ่งแวดล้อม

5.2 ประโยชน์ที่ได้รับ

ระบบตรวจจับและป้องกันฝุ่น PM2.5 นี้ไม่เพียงช่วยเพิ่มความปลอดภัยและสะดวกสบายให้กับผู้ใช้งาน แต่ยังเป็นเครื่องมือสำคัญในการส่งเสริมการป้องกันสุขภาพ มีประโยชน์ดังต่อไปนี้

1. ระบบช่วยให้ผู้ใช้งานทราบค่าฝุ่น PM2.5 ในพื้นที่แบบเรียลไทม์ ซึ่งมีผลต่อการตัดสินใจในการป้องกันตนเอง เช่น การสวมหน้ากาก หรือการหลีกเลี่ยงการทำกิจกรรมกลางแจ้งเมื่อค่าฝุ่นเกินมาตรฐาน

2. การแจ้งเตือนผ่าน LINE Notify ทำให้ผู้ใช้งานรับข้อมูลสำคัญได้ตลอดเวลา ไม่ว่าจะอยู่ที่ใด นอกจากนี้ การควบคุมและแสดงผลค่าฝุ่นผ่านแอปพลิเคชัน Blynk ช่วยเพิ่มความสะดวกในการติดตามค่าฝุ่นแบบเรียลไทม์

3. กราฟแสดงค่าฝุ่น PM2.5 ในแอปพลิเคชัน Blynk ช่วยให้ผู้ใช้งานสามารถวิเคราะห์แนวโน้มของค่าฝุ่นในช่วงเวลาต่างๆ และใช้ข้อมูลดังกล่าวในการวางแผนกิจกรรมประจำวันอย่างเหมาะสม

5.3 ข้อจำกัดของโครงการ

1. การใช้ Blynk บาง Widget บางตัวไม่รองรับในการใช้ฟรี ต้องเสียเงินเติมเพื่อปลดล็อก อาจจะ เป็นสาเหตุที่ต้องจำกัดขอบเขตในการออกแบบ ดีไซน์น้อยลง

2. ข้อจำกัดของเครื่องมือที่ใช้พัฒนา อาจจะยังใช้ไม่ได้กับขอบเขตที่กว้างๆ เลยไม่สามารถ ครอบคลุมได้ทุกจุดอาจจะรองรับแค่บางจุด

3. การแจ้งเตือนผ่าน LINE Notify หรือเสียง Buzzer อาจไม่เหมาะสมในทุกสถานการณ์ เช่น ใน พื้นที่ที่ต้องการความเงียบ หรือในกรณีที่ผู้ใช้งานไม่ได้เปิดใช้งานแอปพลิเคชันที่เกี่ยวข้อง

4. ระบบต้องใช้งานร่วมกับแหล่งจ่ายไฟตลอดเวลา ซึ่งอาจไม่เหมาะสำหรับการติดตั้งในพื้นที่ที่ไม่ มีการเข้าถึงไฟฟ้า

5.4 ข้อเสนอแนะ

1. ปรับปรุงความแม่นยำของเซ็นเซอร์ควรพิจารณาใช้เซ็นเซอร์วัดค่าฝุ่น PM2.5 รุ่นที่มีความ แม่นยำและความทนทานสูงขึ้น เช่น เซ็นเซอร์ที่สามารถตรวจวัดในสภาพแวดล้อมที่มีความชื้นหรือ อุณหภูมิสูง เพื่อให้ค่าที่ได้มีความน่าเชื่อถือมากยิ่งขึ้น

2. ปรับปรุงการแสดงผลข้อมูลควรเพิ่มความหลากหลายในการแสดงผลข้อมูล เช่น การเพิ่ม จอแสดงผลที่มีขนาดใหญ่ขึ้น หรือการสร้างแดชบอร์ดในแอปพลิเคชัน Blynk ให้แสดงข้อมูลที่หลากหลาย เช่น กราฟเปรียบเทียบค่าฝุ่นในช่วงเวลาต่างๆ

บรรณานุกรม

- ระบบควบคุมมลพิษ. (2566). "สถานการณ์ฝุ่นละออง PM2.5 ในประเทศไทย." สืบค้นจาก <https://www.pcd.go.th>
- สำนักงานมาตรฐานผลิตภัณฑ์อุตสาหกรรม (สมอ.). (2563). มาตรฐานผลิตภัณฑ์อุตสาหกรรม มาตรฐานผลิตภัณฑ์อุตสาหกรรม ตรวจวัดฝุ่น PM2.5 (มอก. 2637-2563). กรุงเทพฯ: สมอ.
- สถาบันมาตรวิทยาแห่งชาติ (NIMT). (2564). มาตรฐานการตรวจวัดคุณภาพอากาศในพื้นที่กลางแจ้ง และการสอบเทียบอุปกรณ์วัดฝุ่น PM2.5. กรุงเทพฯ: NIMT.
- สมชาย พัฒนศิริกุล และ สุธีรา เรืองวุฒิ. (2563). "การออกแบบระบบตรวจจับฝุ่น PM2.5 ด้วย เซ็นเซอร์แบบเลเซอร์: กรณีศึกษาพื้นที่เขตเมือง." วารสารวิจัยวิศวกรรมศาสตร์, 35(2), 45-60.
- สมาคมวิศวกรรมสิ่งแวดล้อมแห่งประเทศไทย. (2565). "เทคโนโลยี IoT สำหรับตรวจวัดฝุ่น PM2.5." สืบค้นจาก <https://www.eat.or.th>
- Arduino. (n.d.). เอกสารประกอบแพลตฟอร์ม Blynk: การใช้งานและวิธีติดตั้ง. สืบค้นเมื่อวันที่ 17 ธันวาคม 2567, จาก <https://www.arduino.cc/en/software>
- Ett. (n.d.). PMS5003 PM sensor manual. สืบค้นเมื่อ 17 ธันวาคม 2024, จาก <https://www.ett.co.th>
- Kumar, A., & Gupta, R. (2021). "Air Quality Monitoring Using Low-Cost Sensors and IoT: Challenges and Future Directions." Sensors and Actuators A: Physical, 305(2), 112-125.

ภาคผนวก

ภาคผนวก ก โค้ด ระบบตรวจจับป้องกันฝุ่นPM2.5 และแจ้งเตือนอัตโนมัติ

```
void sendLineNotify(String message) {
  client.setInsecure(); // ปิดการตรวจสอบใบรับรอง SSL

  if (!client.connect("notify-api.line.me", 443)) {
    Serial.println("Failed to connect to LINE Notify server");
    return;
  }

  String body = "message=" + message;
  String header = "POST /api/notify HTTP/1.1\r\n"
    "Host: notify-api.line.me\r\n"
    "Authorization: Bearer " + String(LINE_NOTIFY_TOKEN) + "\r\n"
    "Content-Type: application/x-www-form-urlencoded\r\n"
    "Content-Length: " + String(body.length()) + "\r\n\r\n";

  client.print(header + body);
  Serial.println("Notification sent to LINE!");
  Serial.println("Connecting to LINE Notify server...");

  while (client.available()) {
    String response = client.readString();
    Serial.println("Response: " + response);
  }
}
```

ภาพที่ ก-1 โค้ดฟังก์ชัน การเรียกใช้LineNotify เพื่อเรียกค่าการแจ้งเตือน

```
#define BLYNK_TEMPLATE_ID "TMPL6eW-8Ymo9" // Template ID สำหรับ Blynk
#define BLYNK_TEMPLATE_NAME "Quickstart Template" // ชื่อ Template สำหรับ Blynk
#define BLYNK_AUTH_TOKEN "1sqaXIPKwqq6tEJz4Pv2-klG9bjvabMp" // Blynk Authentication Token
```

ภาพที่ ก-2 โค้ดแสดงการดึงค่าToken ของBlynk มาใช้ในการส่งค่าและควบคุม

```

BLYNK_WRITE(V1) {
  int buzzerCommand = param.asInt();
  if (buzzerCommand == 1) {
    buzzerEnabled = true;
    digitalWrite(BUZZER_PIN, HIGH);
  } else {
    buzzerEnabled = false;
    digitalWrite(BUZZER_PIN, LOW);
  }
}

```

ภาพที่ ก-3 โค้ดการควบคุม เปิด-ปิด ของBuzzer

```

void sendDataToBlynk() {
  pms.wakeUp();
  delay(3000);

  if (pms.readUntil(data)) {
    float pm25 = data.PM_SP_U6_2_5;
    Blynk.virtualWrite(V4, pm25);

    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("PM2.5: ");
    lcd.print(pm25);
    lcd.print(" ug/m3");

    if (pm25 > PM25_ALERT_LEVEL) { // เปรียบเทียบค่า pm25 กับเกณฑ์ PM25_ALERT_LEVEL
      if (!previousAlert) {
        previousAlert = true;
        if (buzzerEnabled) {
          tone(BUZZER_PIN, 1000); // ปรับเสียง Buzzer ให้ทำงาน
        }
        lcd.setCursor(0, 1);
        lcd.print("WARNING!");
        sendLineNotify("PM2.5: " + String(pm25) + " ug/m3. \n\u274C ฝุ่นเกินค่าแล้ว!");
      }
    } else {
      if (previousAlert) {
        previousAlert = false;
        noTone(BUZZER_PIN); // หยุดเสียง Buzzer
        sendLineNotify("PM2.5: " + String(pm25) + " ug/m3. \n✅ ปลอดภัยแล้ว!");
      }
    }
  } else {
    Serial.println("Failed to read PMS5003");
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("Sensor Error");
  }

  pms.sleep();
}

```

ภาพที่ ก-4 โค้ดแสดงการแจ้งเตือนเมื่อค่าฝุ่นมีค่าเกินกว่ากำหนด จะส่งข้อความว่า “ฝุ่นเกินค่าแล้ว”
Buzzer จะส่งเสียงเตือน และเมื่อค่าฝุ่นปกติจะส่งการแจ้งเตือนว่า “ปลอดภัย”

```

void setup() {
  Serial.begin(115200);
  WiFi.begin(ssid, pass);

  Serial.println("Connecting to WiFi...");
  while (WiFi.status() != WL_CONNECTED) {
    delay(1000);
    Serial.print(".");
  }

  Serial.println("\nWiFi connected.");
  Serial.print("IP Address: ");
  Serial.println(WiFi.localIP());

  setupNTP();
}

```

ภาพที่ ก-5 โค้ดสั่งการตั้งค่าWifi ในเข้ากับตัว NodeMCU

```

void sendDataToBlynk() {
  pms.wakeUp();
  delay(3000);

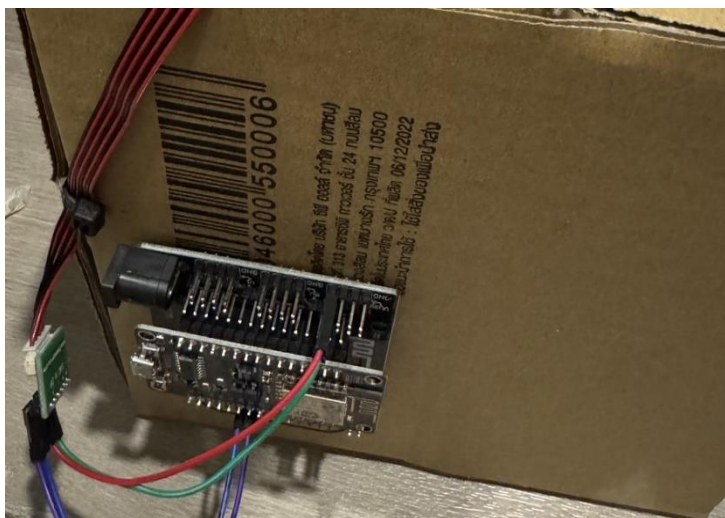
  if (pms.readUntil(data)) {
    float pm25 = data.PM_SP_UG_2_5;
    Blynk.virtualWrite(V4, pm25);

    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("PM2.5: ");
    lcd.print(pm25);
    lcd.print(" ug/m3");
  }
}

```

ภาพที่ ก-6 โค้ดแสดงการส่งค่าของPMS5003 Sensor ส่งไปยังGauge วัดค่าฝุ่นในBlynk

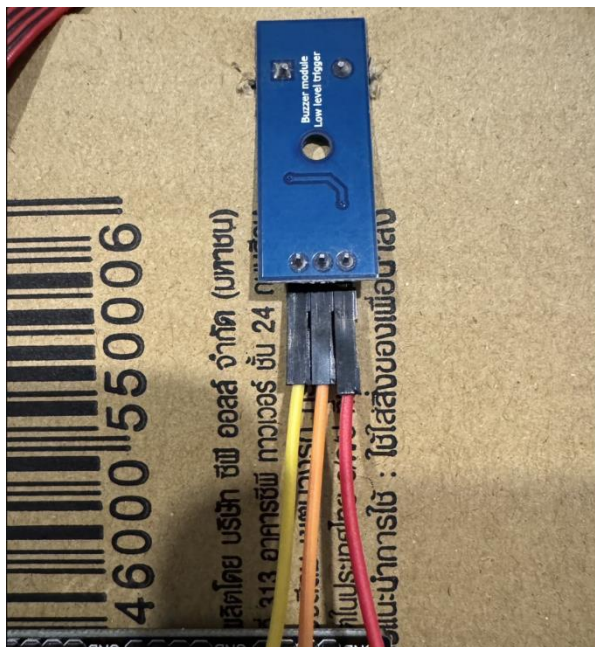
ภาคผนวก ข การทำงาน



ภาพที่ ข-1 การติดตั้ง NodeMCU ESP8266 ลงบนอุปกรณ์



ภาพที่ ข-2 การติดตั้ง PMS5003 Sensor ลงบนอุปกรณ์



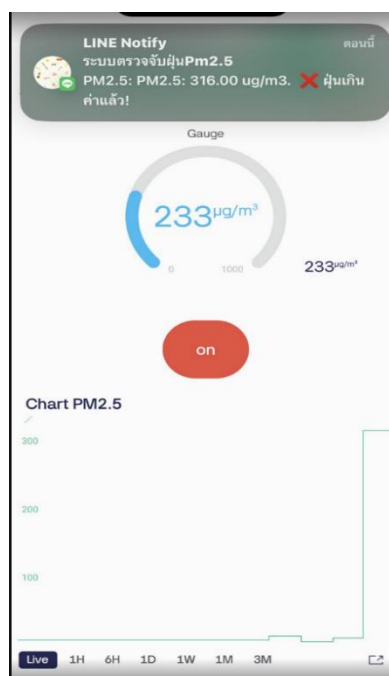
ภาพที่ ข-3 ติดตั้งBuzzer ลงในอุปกรณ์



ภาพที่ ข-4 ติดตั้ง LCD Blacklight ลงบนอุปกรณ์



ภาพที่ ข-5 ติดตั้งเสร็จสมบูรณ์พร้อมใช้งาน



ภาพที่ ข-6 ผลลัพธ์Blynkในควบคุมอุปกรณ์ระบบตรวจจับป้องกันฝุ่นPM2.5 และแจ้งเตือนอัตโนมัติ



ภาพที่ ข-7 ผลลัพธ์ในการแจ้งเตือนค่าฝุ่นPM2.5 ที่เกินกว่ากำหนด และปลอดภัย

ประวัติผู้จัดทำโครงการ

ชื่อ-สกุล	นายชนาธิป รวงผึ้ง
วัน เดือน ปี เกิด	9 สิงหาคม พ.ศ. 2545
ที่อยู่ปัจจุบัน	188/44 ตำบลเสม็ด อำเภอเมือง จังหวัดชลบุรี 20000
อีเมล	64160086@go.buu.ac.th
ประวัติการศึกษา	
ระดับปริญญาตรี	มหาวิทยาลัยบูรพา คณะวิทยาการสารสนเทศ
ระดับมัธยมศึกษาตอนปลาย	โรงเรียนชลราษฎรอำรุง
ระดับมัธยมศึกษาตอนต้น	โรงเรียนชลราษฎรอำรุง
ระดับประถมศึกษา	โรงเรียนปรีชาานุศาสน์
ระดับอนุบาล	โรงเรียน ST.JOSEPH Convent