การประยุกต์ใช้เทคโนโลยีอินเทอร์เน็ตของสรรพสิ่งร่วมกับอุปกรณ์เก็บตัวอย่างฝุ่นตกแบบอัจฉริยะ

จีรวัฒน์ ทองแกมแก้ว

วิทยานิพนธ์เสนอบัณฑิตวิทยาลัย มหาวิทยาลัยนเรศวร
เพื่อเป็นส่วนหนึ่งของการศึกษา หลักสูตรวิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมสิ่งแวดล้อม
ปีการศึกษา 2565
ลิขสิทธิ์เป็นของมหาวิทยาลัยนเรศวร

การประยุกต์ใช้เทคโนโลยีอินเทอร์เน็ตของสรรพสิ่งร่วมกับอุปกรณ์เก็บตัวอย่างฝุ่นตกแบบอัจฉริยะ

จีรวัฒน์ ทองแกมแก้ว

วิทยานิพนธ์เสนอบัณฑิตวิทยาลัย มหาวิทยาลัยนเรศวร
เพื่อเป็นส่วนหนึ่งของการศึกษา หลักสูตรวิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมสิ่งแวดล้อม
ปีการศึกษา 2565
ลิขสิทธิ์เป็นของมหาวิทยาลัยนเรศวร

วิทยานิพนธ์ เรื่อง "การประยุกต์ใช้เทคโนโลยีอินเทอร์เน็ตของสรรพสิ่งร่วมกับอุปกรณ์เก็บตัวอย่างฝุ่นตกแบบอัจฉริยะ"

ของ จีรวัฒน์ ทองแกมแก้ว

ได้รับการพิจารณาให้นับเป็นส่วนหนึ่งของการศึกษาตามหลักสูตร

ปริญญาวิศวกรรมศาสตรมหาบัณฑิต สาขาวิชาวิศวกรรมสิ่งแวดล้อม

คณะกรรมการสอบวิทยานิพนธ์

..... ประธานกรรมการสอบวิทยานิพนธ์

(ดร.วิน ไตรวิทยานุรักษ์)

..... ประธานที่ปรึกษาวิทยานิพนธ์

(ผู้ช่วยศาสตราจารย์ ดร.ปาจรีย์ ทองสนิท)

..... กรรมการผู้ทรงคุณวุฒิภายใน

(ผู้ช่วยศาสตราจารย์ ดร.วิลาวัลย์ คณิตชัยเดชา)

..... กรรมการผู้ทรงคุณวุฒิภายใน

(ดร.สุภาวรรณ ศรีรัตนา)

อนุมัติ

.....

(รองศาสตราจารย์ ดร.กรองกาญจน์ ชูทิพย์)

คณบดีบัณฑิตวิทยาลัย

ชื่อเรื่อง การประยุกต์ใช้เทคโนโลยีอินเทอร์เน็ตของสรรพสิ่งร่วมกับอุปกรณ์เก็บตัวอย่างฝุ่นตกแบบอัจฉริยะ
ผู้วิจัย จีรวัฒน์ ทองแกมแก้ว
ประธานที่ปรึกษา ผู้ช่วยศาสตราจารย์ ดร.ปาจรีย์ ทองสนิท
ประเภทสารนิพนธ์ วิทยานิพนธ์ วศ.ม. วิศวกรรมสิ่งแวดล้อม, มหาวิทยาลัยนเรศวร, 2565
คำสำคัญ อินเทอร์เน็ตของสรรพสิ่ง, อุปกรณ์เก็บตัวอย่างฝุ่นตก, อัจฉริยะ

บทคัดย่อ

งานวิจัยนี้มีวัตถุประสงค์เพื่อประยุกต์ใช้เทคโนโลยีอินเทอร์เน็ตของสรรพสิ่งร่วมกับอุปกรณ์เก็บตัวอย่างฝุ่นตกแบบอัจฉริยะ โดยมีวัตถุประสงค์หลักเพื่อรองรับการใช้งานในช่วงฤดูฝนที่มักมีน้ำฝนตกลงสู่ชุดอุปกรณ์เก็บตัวอย่างเป็นจำนวนมาก สร้างความยากลำบากให้กับผู้ทำการเก็บตัวอย่าง โดยระบบอัจฉริยะที่ถูกติดตั้งเข้าไปเพื่อเพิ่มศักยภาพการทำงานของอุปกรณ์ ได้แก่ ไมโครคอนโทรลเลอร์ เซ็นเซอร์น้ำฝน เซ็นเซอร์วัดความเข้มแสง-อุณหภูมิ-และความชื้น แบบกลางแจ้ง โมดูลขับมอเตอร์ มอเตอร์เกียร์ และระบบโซลาเซลล์แบบออฟกริด ผลการทดสอบพบว่าระบบสามารถทำงานได้อย่างมีประสิทธิภาพ ต่อเนื่อง แม่นยำ สอดคล้องกับสภาวะความเป็นจริงที่เกิดขึ้น โดยเมื่อเกิดสภาวะฝนตกเซ็นเซอร์น้ำฝนจะทำหน้าที่ตรวจจับค่าและสั่งการระบบให้ทำการปิดฝาของโหลเก็บตัวอย่างแบบอัตโนมัติ และสั่งให้ระบบเปิดฝาอีกครั้งเมื่อฝนหยุดลง นอกจากนี้ผู้ควบคุมยังสามารถตรวจค่าสถานะทุกอย่างของระบบผ่านแอปพลิเคชัน Blynk ได้ทุกที่และทุกเวลา

Title THE APPLICATION OF INTERNET OF THINGS (IOT) TECHNOLOGY IN SMART DUST-FALL JAR

Author Chirawat Thongkaemkaeo

Advisor Assistant Professor Pajaree Thongsanit, Ph.D.

Academic Paper M.Eng. Thesis in Environmental Engineering, Naresuan University, 2022

Keywords Internet of Things (IoT), Dust sample collector, Smart/Intelligent

ABSTRACT

This research aims to apply Internet of Things (IoT) technology in smart dust-fall jar to facilitate the usage during rainy season in which a lot of rain falls to smart dust collection devices, make it difficult for dust sample collectors. The smart system shall be installed to increase work performance of the devices, i.e. microcontroller, rain sensor, outdoor light sensor, outdoor temperature sensor, outdoor humidity sensor, motor driver module, gear motor, and off-grid solar system. The test results showed that the system can work efficiently, continuously, and precisely, consistent with a real situation. When it rains, rain sensor shall detect rain and order the system to close the cap of the sample collection bottle automatically and when the rain stops, it shall order the system to open the cap of the sample collection bottle again. Besides, controllers are able to examine all system status through Blynk application from anywhere anytime.

ประกาศคุณูปการ

ผู้วิจัยขอขอบพระคุณเป็นอย่างสูงในความกรุณาของผู้ให้ความช่วยเหลือทุกท่าน ให้คำปรึกษาเป็นอย่างดี ผู้วิจัยขอกราบขอบพระคุณ ผู้ช่วยศาสตราจารย์ ดร.ปาจรีย์ ทองสนิท และ ดร. สุภาวรรณ ศรีรัตนา ที่ปรึกษาวิทยานิพนธ์ ที่กรุณาเป็นที่ปรึกษา พร้อมทั้งแนะนำ แนะแนวทางตลอดระยะเวลาในการทำวิทยานิพนธ์ฉบับนี้ และขอกราบขอบพระคุณ ดร.วิน ไตรวิทยานุรักษ์ ประธานกรรมการสอบวิทยานิพนธ์ ผู้ช่วยศาสตราจารย์ ดร.วิลาวัลย์ คณิตชัยเดชา กรรมการที่ปรึกษาวิทยานิพนธ์ ที่ได้กรุณาให้คำแนะนำตลอดจนแก้ไขข้อบกพร่องของวิทยานิพนธ์ฉบับนี้จนสำเร็จลุล่วงไปด้วยดี

ขอขอบพระคุณ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร ที่ให้ความอนุเคราะห์เงินทุนสนับสนุนวิทยานิพนธ์สำหรับการทำการศึกษาวิจัย อีกทั้งสนับสนุนทางด้านอุปกรณ์ และสถานที่ในการทำวิจัยรวมไปถึงคำปรึกษาที่เป็นประโยชน์จากบุคลากรของภาควิชาทุกท่าน

จีรวัฒน์ ทองแกมแก้ว

สารบัญ

หน้า

บทคัดย่อภาษาไทย ..... ค

บทคัดย่อภาษาอังกฤษ ..... ง

ประกาศคุณูปการ ..... จ

สารบัญ ..... ฉ

สารบัญตาราง ..... ซ

สารบัญภาพ ..... ฌ

บทที่ 1 บทนำ 1

ความเป็นมาของปัญหา 1

จุดมุ่งหมายของการศึกษา 2

ขอบเขตของงานวิจัย 2

นิยามศัพท์เฉพาะ 2

สมมติฐานของการวิจัย 3

บทที่ 2 เอกสารและงานวิจัยที่เกี่ยวข้อง 4

ฝุ่นละออง 4

ผลกระทบของฝุ่นละออง 7

ผลกระทบของฝุ่นละอองที่มีต่อร่างกาย 8

มาตรฐานคุณภาพอากาศ 9

การติดตามตรวจสอบคุณภาพอากาศอย่างง่าย 10

อินเทอร์เน็ตของสรรพสิ่ง (Internet of Things) 11

งานวิจัยที่เกี่ยวข้อง....12
บทที่ 3 วิธีดำเนินงานวิจัย....15
แนวทางในการดำเนินงานวิจัย....15
เครื่องมือที่ใช้ในการวิจัย....16
การสร้างรูปแบบการใช้งานหน้าแอปพลิเคชัน Blynk บนสมาร์ทโฟน....30
ขั้นตอนการทำงานของระบบ....35
การสอบเทียบเซ็นเซอร์วัดความเข้มแสง อุณหภูมิ และความชื้น กลางแจ้ง (Light, Temperature, Humidity outdoor transmitter RS485 output)....36
การติดตั้งและการเก็บข้อมูลการทดสอบอุปกรณ์เก็บตัวอย่างฝุ่นตก....37
บทที่ 4 ผลการวิจัย....39
การเก็บข้อมูลของเซ็นเซอร์ในการวัดค่าอุณหภูมิอากาศ ความชื้นอากาศ และความเข้มแสงอาทิตย์....39
การเปิดโหมดคำสั่งการเปิดปิดฝาโหลเก็บตัวอย่างฝุ่นตกแบบอัตโนมัติ....41
การแสดงผลข้อมูลบนแอปพลิเคชัน Blynk แบบเรียลไทม์....43
บทที่ 5 บทสรุป....44
สรุปและอภิปรายผล....44
ข้อเสนอแนะ....45
บรรณานุกรม....46
ภาคผนวก....49
ประวัติผู้วิจัย....85

สารบัญตาราง

หน้า

ตาราง 1 ส่วนประกอบและแหล่งที่มาของฝุ่นละอองโดยทั่วไป 5

ตาราง 2 ความแตกต่างระหว่างฝุ่นขนาดใหญ่และฝุ่นขนาดเล็ก 6

ตาราง 3 ตัวแปรที่มีอิทธิพลต่อกลไกการตกค้างของฝุ่นละอองในส่วนต่าง ๆ ของระบบหายใจ 8

ตาราง 4 มาตรฐานคุณภาพอากาศในบรรยากาศ (Ambient Air Quality Standards) 9

ตาราง 5 ผลการสอบเทียบเซ็นเซอร์วัดความเข้มแสง อุณหภูมิ และความชื้น กลางแจ้ง (Light, Temperature, Humidity outdoor transmitter RS485 output) 37

สารบัญภาพ

หน้า

ภาพ 1 กรอบแนวความคิดการดำเนินงานวิจัย.....15
ภาพ 2 ESP32-WROOM-32.....16
ภาพ 3 โมดูลขับมอเตอร์ L298N.....17
ภาพ 4 เซ็นเซอร์น้ำฝน ความชื้น.....18
ภาพ 5 โมดูลดิจิตอลเซ็นเซอร์วัดกระแส แรงดัน กำลังไฟฟ้ากระแสตรง.....19
ภาพ 6 ชุดเซ็นเซอร์วัดความเข้มแสง อุณหภูมิ และความชื้น แบบกลางแจ้ง.....20
ภาพ 7 โมดูล MAX485 TTL to RS485.....20
ภาพ 8 การอ่านข้อมูลผ่านพอร์ต Software Serial โดยใช้โมดูล MAX485 TTL to RS485 ..21
ภาพ 9 มอเตอร์เกียร์ 12VDC.....22
ภาพ 10 โมเด็มปล่อยสัญญาณ Wi-Fi.....22
ภาพ 11 การเรียกใช้งานโปรแกรม Arduino IDE.....23
ภาพ 12 การเรียกใช้งานแอปพลิเคชัน Blynk และตัวเลือกการใช้งานต่าง ๆ.....24
ภาพ 13 อุปกรณ์เก็บตัวอย่างฝุ่นตก (Dust-fall Jar) แบบอัจฉริยะ.....25
ภาพ 14 ผังการเชื่อมต่อและการทำงานของอุปกรณ์.....25
ภาพ 15 ตัวอย่างชุดคำสั่งส่วนของ Header.....27
ภาพ 16 ตัวอย่างชุดคำสั่งส่วนของ Setup.....28
ภาพ 17 ตัวอย่างชุดคำสั่งส่วนของ Loop.....29
ภาพ 18 หน้า Dashboard การใช้งานบนสมาร์ทโฟนในแอปพลิเคชัน Blynk.....30
ภาพ 19 หน้า Dashboard การปรับค่าการทำงานของมอเตอร์ในโหมด auto.....31

ภาพ 20 การลงทะเบียนใช้งานแอปพลิเคชัน Blynk.....32

ภาพ 21 การเชื่อมต่อสมาร์ทโฟนกับอุปกรณ์ผ่าน Wi-Fi ในแอปพลิเคชัน Blynk.....33

ภาพ 22 การสร้างหน้า Dashboard ในแอปพลิเคชัน Blynk.....34

ภาพ 23 การตั้งค่าการรับข้อมูลในส่วน Widget.....34

ภาพ 24 แผนผังขั้นตอนการทำงานของระบบ.....35

ภาพ 25 การเปรียบเทียบกับข้อมูลจากเว็บไซต์กรมอุตุนิยมวิทยาพิษณุโลก.....36

ภาพ 26 ตำแหน่งการติดตั้งอุปกรณ์เก็บตัวอย่างฝุ่นตก.....37

ภาพ 27 อุณหภูมิอากาศ ความชื้นอากาศ และความเข้มแสงอาทิตย์ บริเวณจุดติดตั้งอุปกรณ์วันที่ 28 กันยายน พ.ศ. 2565 ถึงวันที่ 2 ตุลาคม พ.ศ. 2565 บันทึกข้อมูลทุกๆ 1 นาที เป็นเวลา 24 ชั่วโมง.....40

ภาพ 28 สถานะของการเปิดปิดฝาโหลเก็บตัวอย่างฝุ่นตก อุณหภูมิอากาศ ความชื้นอากาศ และเซ็นเซอร์น้ำฝน บริเวณจุดติดตั้งอุปกรณ์ วันที่ 3 ตุลาคม พ.ศ. 2565 ถึงวันที่ 6 ตุลาคม พ.ศ. 2565.....42

ภาพ 29 การแสดงค่าของเซ็นเซอร์แบบเรียลไทม์บนแอปพลิเคชัน Blynk.....43

บทที่ 1

บทนำ

ความเป็นมาของปัญหา

ปัจจุบันโลกได้เข้าสู่ยุคที่การเข้าถึงข้อมูลต่าง ๆ และการติดต่อสื่อสารทำได้อย่างรวดเร็ว รวมถึงการใช้เทคโนโลยีที่ทันสมัยเพื่ออำนวยความสะดวกสบายของมนุษย์ หลายสิ่งล้วนใช้งานผ่านเครือข่ายอินเทอร์เน็ต ยกตัวอย่าง เช่น การสนทนาแบบเห็นหน้าผ่านสมาร์ทโฟน การประชุมแบบออนไลน์ การสั่งซื้อสินค้าและบริการผ่านแพลตฟอร์ม การทำธุรกรรมการเงิน การรับส่งข้อมูลแบบเรียลไทม์ เป็นต้น จึงปฏิเสธไม่ได้ว่าอินเทอร์เน็ตได้เข้ามามีบทบาทในการใช้ชีวิตประจำวันเป็นอย่างมาก เพื่อตอบสนองความต้องการของมนุษย์ที่ต้องการความรวดเร็วและความสะดวกสบาย อิทธิพลของการใช้เครือข่ายอินเทอร์เน็ตส่งผลให้เทคโนโลยีต่างต้องปรับตัวเพื่อประยุกต์การใช้งานร่วมกัน การใช้อินเทอร์เน็ตร่วมกับเทคโนโลยีช่วยให้การวิเคราะห์ข้อมูล การตรวจสอบสถานะการทำงาน และการวิเคราะห์ปัญหาทำได้อย่างรวดเร็วและช่วยให้มีประสิทธิภาพมากยิ่งขึ้น

อินเทอร์เน็ตของสรรพสิ่ง (Internet of Things) หรือ ไอโอที (IoT) คือ อุปกรณ์ สิ่งของเครื่องใช้ พาหนะ และสิ่งอำนวยความสะดวกที่มนุษย์สร้างขึ้น โดยมีการติดตั้งอุปกรณ์อิเล็กทรอนิกส์ เซ็นเซอร์ และซอฟต์แวร์ ใช้งานร่วมกับเครือข่ายอินเทอร์เน็ต ส่งผลให้อุปกรณ์สามารถจัดเก็บหรือรับส่งข้อมูลแบบเรียลไทม์ได้ รวมถึงการใช้เซ็นเซอร์ตรวจสอบสภาพแวดล้อมและควบคุมอุปกรณ์ได้จากระยะไกล ไอโอทีช่วยให้อุปกรณ์มีประสิทธิภาพ ความแม่นยำ และความสะดวกในการใช้งานเพิ่มมากขึ้น สามารถนำมาประยุกต์ใช้กับงานได้หลายด้าน เช่น ด้านพลังงาน ด้านวิศวกรรม ด้านการสื่อสาร ด้านการคมนาคม เป็นต้น

การตรวจวัดฝุ่นละอองในบรรยากาศโดยทั่วไป มีการใช้อุปกรณ์เก็บตัวอย่างฝุ่นละอองในบรรยากาศเพื่อติดตามตรวจสอบคุณภาพอากาศ การเลือกใช้อุปกรณ์เก็บตัวอย่างมีหลากหลายเทคนิคขึ้นอยู่กับขนาดของฝุ่นละออง การติดตามตรวจสอบคุณภาพอากาศอย่างง่ายโดยใช้อุปกรณ์เก็บตัวอย่างฝุ่นตก (Dust fall Jar) วิธีนี้เป็นการเก็บรวบรวมฝุ่นละอองแขวนลอยในบรรยากาศที่ตกลงตามแรงโน้มถ่วงของโลก อุปกรณ์นี้สามารถเก็บตัวอย่างฝุ่นละอองได้โดยไม่ต้องอาศัยการไหลของอากาศผ่านอุปกรณ์ โดยขนาดของฝุ่นละอองที่อุปกรณ์นี้สามารถเก็บได้มีตั้งแต่ขนาดใหญ่กว่า 100 ไมครอน ถึงขนาดเล็กสุด 20 – 50 ไมครอน ซึ่งข้อจำกัดของอุปกรณ์เก็บตัวอย่างฝุ่นตก คือ ไม่สามารถทำการตรวจวัดในช่วงเวลาที่เกิดฝนตกได้เพราะอาจทำให้เกิดน้ำขังภายในโหลเก็บตัวอย่างฝุ่นตกได้ ซึ่งส่งผลต่อความแม่นยำในการเก็บตัวอย่างฝุ่นตกและการวิเคราะห์ผล จากข้อจำกัดข้างต้นจึงมี

การพัฒนาอุปกรณ์เก็บตัวอย่างฝุ่นตกแบบป้องกันฝนอัตโนมัติโดยใช้พลังงานจากเซลล์แสงอาทิตย์ ซึ่งอุปกรณ์ดังกล่าวได้มีการพัฒนาระบบเปิดปิดฝาครอบโหลเก็บตัวอย่างฝุ่นตก เพื่อป้องกันน้ำฝนที่เข้าไปขังภายใน มีหลักการทำงาน คือ เมื่อฝนตกเซ็นเซอร์วัดฝนจะส่งข้อมูลให้กับไมโครคอนโทรลเลอร์เพื่อควบคุมมอเตอร์ในการเปิดปิดฝาครอบโหลเก็บตัวอย่างฝุ่นตกโดยอัตโนมัติ

ในการพัฒนาอุปกรณ์เก็บตัวอย่างฝุ่นตกนี้ได้ใช้เซ็นเซอร์และอุปกรณ์อิเล็กทรอนิกส์ในการควบคุมการทำงานของอุปกรณ์ ผู้วิจัยจึงเล็งเห็นการประยุกต์ใช้งานอุปกรณ์ดังกล่าวร่วมกับเครือข่ายอินเทอร์เน็ต เพื่อให้การพัฒนามีประสิทธิภาพและความสะดวกในการทำงานมากขึ้น งานวิจัยนี้มีวัตถุประสงค์เพื่อประยุกต์ใช้เทคโนโลยีอินเทอร์เน็ตของสรรพสิ่งร่วมกับอุปกรณ์เก็บตัวอย่างฝุ่นตกแบบอัจฉริยะ โดยทำการติดตั้งโมดูลวัดค่ากระแสไฟฟ้า เซ็นเซอร์วัดค่าพารามิเตอร์ การสั่งการระยะไกลผ่านแอปพลิเคชันใช้งานร่วมกับเครือข่ายอินเทอร์เน็ตและแสดงผลข้อมูลสภาพแวดล้อมแบบเรียลไทม์

จุดมุ่งหมายของการศึกษา

1. เพื่อประยุกต์ใช้เทคโนโลยีอินเทอร์เน็ตของสรรพสิ่งร่วมกับอุปกรณ์เก็บตัวอย่างฝุ่นตกแบบอัจฉริยะ ให้อุปกรณ์เก็บตัวอย่างฝุ่นตกมีความสะดวกในการใช้งานมากยิ่งขึ้น
2. เพื่อทดสอบประสิทธิภาพการทำงานของระบบเทคโนโลยีอินเทอร์เน็ตของสรรพสิ่งในการควบคุมการทำงานของอุปกรณ์เก็บตัวอย่างฝุ่นตก

ขอบเขตของงานวิจัย

งานวิจัยนี้ทำการทดสอบการทำงานของเครื่องเก็บตัวอย่างฝุ่นตกและความแม่นยำในการทำงานของเซ็นเซอร์ โดยติดตั้งโมดูลวัดค่ากระแสไฟฟ้า เซ็นเซอร์ตรวจวัดค่าอุณหภูมิ ความชื้น ความเข้มแสง เพื่อส่งค่าแสดงข้อมูลแบบเรียลไทม์และทดสอบความแม่นยำในการสั่งการเปิดปิดระยะไกลผ่านแอปพิลเคชัน ซึ่งทำการทดลองในพื้นที่ตำบลหัวรอ อำเภอเมือง จังหวัดพิษณุโลก

นิยามศัพท์เฉพาะ

1. อินเทอร์เน็ตของสรรพสิ่ง (Internet of Things) หรือ ไอโอที (IoT) หมายถึง อุปกรณ์ สิ่งของเครื่องใช้ พาหนะ และสิ่งอำนวยความสะดวกที่มนุษย์สร้างขึ้น โดยมีการติดตั้งอุปกรณ์อิเล็กทรอนิกส์ เซ็นเซอร์ และซอฟต์แวร์ ใช้งานร่วมกับเครือข่ายอินเทอร์เน็ต
2. อุปกรณ์เก็บตัวอย่างฝุ่นตก (Dust-fall Jar) หมายถึง ภาชนะเก็บตัวอย่างฝุ่นละอองแขวนลอยในบรรยากาศที่ตกลงตามแรงโน้มถ่วงของโลก โดยไม่อาศัยการไหลของอากาศผ่านอุปกรณ์

เก็บตัวอย่างฝุ่น สามารถเก็บตัวอย่างฝุ่นตกที่มีอนุภาคขนาดใหญ่กว่า 100 ไมครอน ถึงขนาดเล็กสุด 20 – 50 ไมครอน

สมมติฐานของการวิจัย

อุปกรณ์เก็บตัวอย่างฝุ่นตกสามารถแสดงข้อมูลพารามิเตอร์ของสิ่งแวดล้อมในบริเวณพื้นที่เก็บข้อมูล ค่ากระแสไฟฟ้า และสามารถสั่งการเปิดปิดอุปกรณ์ระยะไกลผ่านแอปพิลเคชันที่ใช้งานร่วมกับเครือข่ายอินเทอร์เน็ตได้อย่างมีประสิทธิภาพ

บทที่ 2

เอกสารและงานวิจัยที่เกี่ยวข้อง

ฝุ่นละออง

ฝุ่นละออง คือ อนุภาคของแข็งหรือของเหลวที่มีอยู่ในอากาศ ฝุ่นละอองเกิดจากกิจกรรมต่าง ๆ ทั้งจากธรรมชาติและมนุษย์กระทำ ซึ่งอาจเกิดจาก การจราจร กิจกรรมทางอุตสาหกรรม และการเผาไหม้ที่โล่งแจ้ง ซึ่งทำให้องค์ประกอบทางกายภาพและเคมีของฝุ่นมีความหลากหลาย ที่มีผลมาจากการรวมตัวของอนุภาค ฝุ่นและก๊าซบางชนิด เช่น ซัลเฟอร์ไดออกไซด์ออกไซด์ของไนโตรเจน เป็นต้น ฝุ่นละอองในบรรยากาศสามารถแบ่งออกได้เป็น 2 ประเภท คือ ฝุ่นปฐมภูมิ ซึ่งเกิดขึ้นและกระจายสู่ชั้นบรรยากาศโดยตรง และฝุ่นทุติยภูมิ เกิดจากปฏิกิริยาต่าง ๆ ในบรรยากาศ เช่น การควบแน่นของฝุ่นเข้าด้วยกัน การรวมตัวของฝุ่นและก๊าซที่ควบแน่นของฝุ่นกับของเหลวหรือการควบแน่นด้วยของแข็งฝุ่นละอองทั่วไปในบรรยากาศมีขนาดใหญ่กว่า 500 ไมครอน ถึงขนาดเล็กที่สุดถึง 0.002 ไมครอน

ฝุ่นละอองขนาดใหญ่ เกิดจากกิจกรรมการก่อสร้างต่าง ๆ ซึ่งทำให้เกิดการฟุ้งกระจายของดินหรือทรายในอากาศและตกลงสู่พื้นอย่างรวดเร็ว

ฝุ่นละอองแขวนลอยในบรรยากาศ คือ ฝุ่นละอองที่สามารถแขวนลอยอยู่ในอากาศได้เป็นเวลานานมีขนาดเล็กกว่าหรือเท่ากับ 100 ไมครอน เช่น การเผาในที่โล่ง ควันจากท่อไอเสียยานพาหนะ จากกิจกรรมของโรงงานอุตสาหกรรม เป็นต้น

ฝุ่นละอองขนาดเล็ก เกิดจากการเผาไหมเป็นส่วนใหญ่ เช่น การสูบบุหรี่ การเผาไหม้ของเครื่องยนต์ดีเซล การเผาในที่โล่ง และปฏิกิริยาเคมีในอากาศ โดยแบ่งฝุ่นละอองตามขนาดออกเป็น 3 ชนิดได้ดังนี้

จากการศึกษาวิจัยพบว่าฝุ่นที่ทำให้เกิดปัญหาสุขภาพ จะเป็นฝุ่นขนาดเล็กที่มีขนาดไม่เกิน 10 ไมครอน โดยฝุ่นขนาดเล็กนี้จะสามารถเข้าไปในทางเดินหายใจผ่านโพรงจมูกไปยังถุงลมในปอด และฝุ่นจะเป็นพิษมากขึ้นหากฝุ่นเกิดจากการควบแน่นของก๊าซบางชนิด เช่น ซัลเฟอร์ไดออกไซด์ ไนโตรเจนออกไซด์ การอยู่ในบริเวณที่มีฝุ่นละอองในปริมาณมากหากได้รับการสัมผัสทำให้เสี่ยงต่อโรคในระบบทางเดินหายใจมากขึ้น (สำนักจัดการคุณภาพอากาศและเสียง, 2554)

การกำหนดมาตรฐานฝุ่นละอองในบรรยากาศของประเทศไทย ได้กำหนดมาตรฐานฝุ่นละอองขนาดสูงถึง 100 ไมครอน หรือ TSP ขนาดฝุ่นสูงสุด 10 ไมครอน หรือ PM10 และขนาดฝุ่นสูงสุด 2.5 ไมครอน หรือ PM2.5 ซึ่งจะมีค่ามาตรฐานเฉลี่ย 24 ชั่วโมงและ 1 ปี สำหรับ TSP มีค่ามาตรฐาน เฉลี่ย 24 ชั่วโมง ไม่เกิน 330 ไมโครกรัมต่อลูกบาศก์เมตร และ มาตรฐานเฉลี่ย 1 ปี ไม่เกิน 100 ไมโครกรัมต่อลูกบาศก์เมตร PM10 เฉลี่ย 24 ชั่วโมง ไม่เกิน 120 ไมโครกรัมต่อลูกบาศก์เมตร และ มาตรฐานเฉลี่ย 1 ปี ไม่เกิน 50 ไมโครกรัมต่อลูกบาศก์เมตร และ PM2.5 มีค่าเฉลี่ย 24 ชั่วโมง ไม่เกิน 50 ไมโครกรัมต่อลูกบาศก์เมตร และมาตรฐานเฉลี่ย 1 ปี ไม่เกิน 25 ไมโครกรัมต่อลูกบาศก์เมตร (สำนักจัดการคุณภาพอากาศและเสียง, 2554)

ตาราง 1 ส่วนประกอบและแหล่งที่มาของฝุ่นละอองโดยทั่วไป

ส่วนประกอบ	แหล่งที่มา
สารประกอบอินทรีย์ เช่น ไดออกซิน ไดแบนโซฟู	ผลผลิตทางเคมีจากกระบวนการผลิตสารคลอโรฟินอล
แรน โพลีไซคลิก	คลอโรเบนซิน และสารกำจัดวัชพืช
แอโรแมติกไฮโดรคาร์บอน (PAH)	การเผาชีวมวล ไฟป่า และกระบวนการเผาไหม้ที่ไม่
	สมบูรณ์ในเครื่องยนต์ดีเซลและเบนซิน
แอมโมเนีย	การผลิตและการใช้ของเสียจากสัตว์และปุ๋ย
แคลเซียมซัลเฟต	วัสดุที่ใช้ในงานก่อสร้าง เช่น ยิปซัม
ซัลเฟอร์ไดออกไซด์	การเผาไหม้ของเชื้อเพลิง โดยเฉพาะถ่านหิน น้ำมัน
ตะกั่ว	น้ำมันที่มีสารตะกั่ว
ดิน	แร่ธาตุต่าง ๆ

ที่มา: บวร ไชยษา, 2546

ตาราง 2 ความแตกต่างระหว่างฝุ่นขนาดใหญ่และฝุ่นขนาดเล็ก

	ฝุ่นขนาดใหญ่	ฝุ่นขนาดเล็ก
แหล่งที่มา	- การฟุ้งกระจายของดินบนถนน - การฟุ้งของฝุ่นดินที่เกิดจากการทำเหมืองแร่ สัตว์เลี้ยง - เศษสิ่งมีชีวิต - การก่อสร้างและรื้อถอน - การเผาไหม้ของถ่านหินและน้ำมัน - ทะเล มหาสมุทร	- การเผาไหม้ถ่านหิน น้ำมัน เศษไม้ - การเปลี่ยนสภาพของก๊าซไนโตรเจนออกไซด์ ซัลเฟอร์ไดออกไซด์ และสารประกอบอินทรีย์ในบรรยากาศ - กรบวนการที่ใช้ความร้อนสูง เช่น เตาหลอมโรงบดเหล็ก เป็นต้น
กระบวนการ	- ถูกบด กระแทก - การระเหยของแก๊สบางชนิด - การแขวนลอยของผงฝุ่น	- กระบวนการทางเคมีกลายเป็นไอ - Nucleation, Condensation และ Coagulation - การระเหยของหมอก และหยดน้ำในก้อนเมฆซึ่งมีก๊าซละลายและเกิดปฏิกิริยา
องค์ประกอบหลัก	- ผงฝุ่นที่ฟุ้งกระจาย - ขี้เถ้าลอยจากถ่านหินและน้ำมัน - ออกไซด์ของธาตุที่เป็นองค์ประกอบของเปลือกโลก - $CaCo_3$, NaCl, ฝุ่นจากเกลือทะเล - เกสรดอกไม้ สปอร์ของเชื้อรา - ฝุ่นที่เกิดจากยางรถยนต์	- ซัลเฟต (SO_4) - ไนเตรท (NO_3) - แอมโมเนีย (NH_4^+) - ไฮโดรเจนอิออน (H) - ธาตุคาร์บอน (C) - คาร์บอนอินทรีย์ (eg.,PAHS) - โลหะ (Pb, Cd, Ni, Cu, Zn) - ละอองน้ำที่จับตัวกับฝุ่น

ที่มา: นริศรา ไทยนาม, 2552

ผลกระทบของฝุ่นละออง

1. ผลกระทบต่อฝุ่นละอองทั่วไป

ฝุ่นละอองในอากาศสามารถดูดซับและหักเหแสงได้สงผลให้ความสามารถในการมองเห็นลดลงและเกิดทัศนวิสัยในการมองเห็นที่ไม่ดี หากมีฝุ่นละอองแขวนลอยในอากาศมากจนกลายเป็นหมอก เป็นอุปสรรคต่อการมองเห็นอาจทำให้เกิดอุบัติเหตุในการสัญจรได้ ฝุ่นละอองมีส่วนช่วยเร่งปฏิกิริยาทำให้เกิดภาวะมลพิษทางอากาศที่รุนแรง โดยเฉพาะการเกิดการรวมตัวกับซัลเฟอร์ไดออกไซด์ในอากาศ เกิดเป็นกรดซัลฟุริกที่มีอันตรายต่อระบบทางเดินหายใจ (ศลีจิตร น้ำจิตร และคณะ, 2545)

2. ผลกระทบต่อวัตถุและสิ่งก่อสร้าง

ฝุ่นละอองในอากาศที่ตกลงตามแรงดึงดูดของโลก หากเกาะติดกับวัตถุและสิ่งก่อสร้างนอกจากทำให้สกปรกแล้ว ยังมีสมบัติในการดูดซับโลหะ สารอินทรีย์และสารอนินทรีย์ไว้ที่ผิวของฝุ่นละอองด้วยหรือจากชนิดของฝุ่นละอองที่มีสภาพเป็นกรดหรือมีองค์ประกอบทางเคมีที่เป็นอันตราย เมื่อเกาะติดกับวัตถุหรือสิ่งก่อสร้างจะทำอันตรายต่อสิ่งนั้น เช่น ทำให้วัสดุสีกร่อน ทำลายผิวหน้าของสิ่งก่อสร้าง ผลงานทางศิลปะเสื่อมสภาพ และหลังคาสังกะสีผุกร่อน เป็นต้น (ศลีจิตร น้ำจิตร และคณะ, 2545)

3. ผลกระทบต่อพืช

เมื่อฝุ่นตกลงมาสัมผัสกับพืช ฝุ่นจะจับติดส่วนต่าง ๆ ของพืช โดยเฉพาะใบซึ่งเป็นส่วนที่มีพื้นผิวมากและรับการตกของฝุ่นได้ดี ส่งผลให้ประสิทธิภาพการสังเคราะห์แสงลดลง ฝุ่นที่ปิดบริเวณปากใบทำให้เกิดการสะสมความร้อนไว้ภายในมากขึ้น ส่งผลต่อการเจริญเติบโตของพืชถ้าหากฝุ่นละอองปนเปื้อนสารพิษและตกลงมาเกาะที่พืชก็ทำให้พืชนั้นปนเปื้อนพิษได้ (ศลีจิตร น้ำจิตร และคณะ, 2545)

4. ผลกระทบต่อสุขภาพของมนุษย์

ฝุ่นละอองสามารถส่งผลให้เสียชีวิตก่อนวัยอันควร เนื่องจากทำให้เกิดโรคระบบทางเดินหายใจและโรคในระบบหัวใจและหลอดเลือด ซึ่งระดับความรุนแรงของอาการป่วยจะเปลี่ยนแปลงตามระดับของฝุ่นละออง จากการศึกษาพบว่าอัตราการเข้ารักษาตัวในโรงพยาบาลด้วยโรคระบบทางเดินหายใจ โรคหัวใจและหลอดเลือดเพิ่มสูงขึ้นเมื่อมีฝุ่นละอองขนาดเล็กกว่า 10 ไมครอน ในอากาศมีปริมาณมากและมีโอกาสป่วยมากขึ้นในสถานที่ที่ไม่ใช้เครื่องปรับอากาศ (ศลีจิตร น้ำจิตร และคณะ, 2545)

ผลกระทบของฝุ่นละอองที่มีต่อร่างกาย

1. ระยะฝุ่นภายนอกร่างกาย

ฝุ่นละอองที่มีขนาด 0.4 – 0.9 ไมครอน ซึ่งแขวนลอยอยู่ในอากาศสามารถปิดกั้นการเดินทางของแสงได้ สงผลให้การมองเห็นในระยะไกลไม่ชัดเจน ส่วนฝุ่นละอองที่มีขนาดใหญ่ทำให้เกิดการระคายเคืองต่อระบบประสาทรับความรู้สึก เช่น ตา จมูก และคอ เป็นต้น (ศลีจิตร น้ำจิตร และคณะ, 2545)

2. ระยะฝุ่นเมื่อเข้าสู่ร่างกาย

ฝุ่นละอองที่มีขนาดเล็กกว่า 10 ไมครอน สามารถเข้าสู่ระบบทางเดินหายใจได้ โดยเมื่อฝุ่นละอองขนาดเล็กถูกสูดเข้าสู่ร่างกายด้วยความเร็วลมจากการหายใจเข้าจะสัมผัสกับส่วนต่าง ๆ ของหลอดลมและจะถูกแรงโน้มถ่วงพาให้ตกลงสู่ถุงลมปอด จากนั้นฝุ่นละอองอาจถูกขับออกโดยกลไกของร่างกาย เช่น เมื่อมีอัตราการหายใจสูงหรือหายใจแรง ฝุ่นละอองจะออกมากับลมหายใจหรืออาจติดค้างอยู่ในปอด หากฝุ่นละอองติดค้างอยู่ในระบบหายใจมาก ๆ เป็นเวลานานส่งผลให้เกิดโรคระบบทางเดินหายใจ (ศลีจิตร น้ำจิตร และคณะ, 2545)

ตาราง 3 ตัวแปรที่มีอิทธิพลต่อกลไกการตกค้างของฝุ่นละอองในส่วนต่าง ๆ ของระบบหายใจ

ขนาดของฝุ่นละออง	กลไกและบริเวณตกค้างของฝุ่นละอองในทางเดินหายใจ
5-00 ไมครอน	การปะทะเหตุความเฉื่อย
	จมูกและคอหอยส่วนจมูก
1-10 ไมครอน	การตกตะกอน
	คอหอยและหลอดลม
	หลอดลมและหลอดลมฝอย
1 ไมโครเมตรและเล็กกว่า	การแผ่ซ่าน
	ถุงลม
	บริเวณถุงลม

ที่มา: บวร ไชยษา, 2546

มาตรฐานคุณภาพอากาศ

เป็นการกำหนดระดับความเข้มข้นของฝุ่นละอองในบรรยากาศสูงสุดซึ่งยินยอมให้มีได้ในบรรยากาศตามกฎหมาย เพื่อป้องกันมิให้เกิดอันตรายต่อประชาชนหรือระบบนิเวศน์ ซึ่งประเทศ ไทยได้จัดทำมาตรฐานคุณภาพอากาศในบรรยากาศตั้งแต่ปี พ.ศ. 2524 ตาม พระราชบัญญัติ ส่งเสริมและรักษาคุณภาพสิ่งแวดล้อมปี พ.ศ. 2518 ซึ่งได้มีการกำหนดค่าความเข้มข้นฝุ่นละออง (Total Suspended particulates) ในบรรยากาศค่าเฉลี่ยในเวลา 24 ชั่วโมง มีค่าไม่เกิน 0.33 มิลลิกรัม/ลูกบาศก์เมตร และค่าเฉลี่ย 1 ปีมีค่าไม่เกิน 0.1 มิลลิกรัม/ลูกบาศก์เมตร ซึ่งเป็นค่าเฉลี่ย ทางเรขาคณิต (Geometric mean) โดยใช้วิธีวัดแบบการชั่งน้ำหนัก (Gravimetric method)

ต่อมาได้มีการจัดทำมาตรฐานคุณภาพอากาศในบรรยากาศขึ้นใหม่ในปี พ.ศ. 2538 ตามพระราชบัญญัติส่งเสริมและรักษาคุณภาพสิ่งแวดล้อมปี พ.ศ. 2535 โดยกรมควบคุมมลพิษ สำนักนโยบายและแผนสิ่งแวดล้อม กระทรวงวิทยาศาสตร์ เทคโนโลยีและสิ่งแวดล้อม ได้มีการกำหนดค่าความเข้มข้นของฝุ่นละอองในบรรยากาศ โดยใช้วิธีวัดแบบ Gravimetric-High Volume ได้แบ่งออกเป็น 2 ขนาด คือ 1. ฝุ่นรวม (TSP) มีค่าความเข้มข้นมาตรฐานในบรรยากาศเฉลี่ย 24 ชั่วโมง ไม่เกิน 0.33 มิลลิกรัมต่อลูกบาศก์เมตร และ ค่าเฉลี่ยใน 1 ปี มีค่าไม่เกิน 0.1 มิลลิกรัมต่อลูกบาศก์เมตร และ 2. ฝุ่นขนาดเล็กกว่า 10 ไมครอน (PM10) กำหนดให้มีค่าความเข้มข้นใน บรรยากาศ เฉลี่ย 24 ชั่วโมง ไม่เกิน 0.12 มิลลิกรัมต่อลูกบาศก์เมตร และค่าเฉลี่ยในเวลา 1 ปี มีค่า ไม่เกิน 0.05 มิลลิกรัมต่อลูกบาศก์เมตร (ขวัญฤทัย ทองบุญฤทธิ์, 2554)

ตาราง 4 มาตรฐานคุณภาพอากาศในบรรยากาศ (Ambient Air Quality Standards)

สารมลพิษ	ค่าเฉลี่ย 1 ชั่วโมง		ค่าเฉลี่ย 8 ชั่วโมง		ค่าเฉลี่ย 24 ชั่วโมง		ค่าเฉลี่ย 1 เดือน		ค่าเฉลี่ย 1 ปี		วิธีการตรวจวัด
	mg/m^3	ppm	mg/m^3	ppm	mg/m^3	ppm	mg/m^3	ppm	mg/m^3	ppm	
คาร์บอนมอนอกไซด์ (CO)	34.20	30	10.26	9	-	-	-	-	-	-	Non-Dispersive Infrared Detection
ไนโตรเจนไดออกไซด์ (NO_2)	0.32	0.17	-	-	-	-	-	-	-	-	Chemiluminescence
ซัลเฟอร์ไดออกไซด์(SO_2)	0.78	0.30	-	-	0.30	0.12	-	-	0.10	0.04	Pararosaniline
ฝุ่นละอองรวม (TSP)	-	-	-	-	0.33	-	-	-	0.10	-	Gravimetric-High Volume
ฝุ่นขนาดเล็กกว่า 10ไมครอน (PM-10)	-	-	-	-	0.12	-	-	-	0.05	-	Gravimetric-High Volume

สารมลพิษ	ค่าเฉลี่ย 1 ชั่วโมง		ค่าเฉลี่ย 8 ชั่วโมง		ค่าเฉลี่ย 24 ชั่วโมง		ค่าเฉลี่ย 1 เดือน		ค่าเฉลี่ย 1 ปี		วิธีการตรวจวัด
	mg/m^3	ppm	mg/m^3	ppm	mg/m^3	ppm	mg/m^3	ppm	mg/m^3	ppm	
โอโซน (O_3)	0.20	0.10	-	-	-	-	-	-	-	-	Chemiluminescence
ตะกั่ว (Pb)	-	-	-	-	-	-	1.5	-	-	-	Atomic Absorption Spectrometer
ก๊าซคาร์บอนไดซัลไฟด์ (CS_2)	-	-	-	-	0.18	-	-	-	-	-	US EPA Compendium Method TO-15

หมายเหตุ: 1) ค่าเฉลี่ยทางเรขาคณิต

2) ความเข้มข้นของก๊าซแต่ละชนิดในบรรยากาศที่ 1 บรรยากาศ 25 °C

3) กำหนดมาตรฐานเฉลี่ยระยะสั้น (1, 8 และ 24 ซม.) เพื่อป้องกันผลกระทบเฉียบพลันต่อสุขภาพ

4) มาตรฐานเฉลี่ยระยะยาว (1 เดือนและ 1 ปี) กำหนดขึ้นเพื่อป้องกันผลกระทบระยะยาว ผลเรื้อรังที่อาจเกิดขึ้นต่อสุขภาพ (chronic effect)

ที่มา: ยุทธนา ตันวงศ์วาล, 2560

การติดตามตรวจสอบคุณภาพอากาศอย่างง่าย

การตรวจวัดฝุ่นตกโดยการใช้ภาชนะเก็บฝุ่นตก (Dust-fall jar) คือ การเก็บรวบรวมอนุภาคที่ตกลงจากบรรยากาศด้วยแรงโน้มถ่วงของโลก โดยวิธีการเก็บตัวอย่างนี้ไม่ต้องอาศัยแหล่งสุญญากาศหรือระบบตวงวัดปริมาณการไหล สามารถเก็บตัวอย่างได้โดยใช้ภาชนะเก็บฝุ่นตก (Dust fall jar Container) วิธีนี้เหมาะสำหรับอนุภาคขนาดใหญ่จนกระทั่งขอนาดเล็กสุด 20 – 50 ไมครอน สามารถคำนวณหาปริมาณฝุ่นตกได้ในหน่วย $mg/m^2/day$ โดยมีวิธีเก็บตัวอย่างดังนี้

1. อยู่ห่างจากแหล่งกำเนิดมลพิษต่าง ๆ อย่างน้อย 50 เมตร
2. ตามแนวราบโดยรอบไม่มีกำแพงหรือสิ่งกีดขวางอื่นใดอย่างน้อย 10 เมตร
3. สูงจากพื้นอย่างน้อย 1.5 เมตร

โดยปกติจะวางอุปกรณ์ไว้ตลอดช่วง 30 วัน ควรมีการจดบันทึกสภาพอากาศจากนั้นเก็บโหลเก็บตัวอย่างไปวิเคราะห์ต่อในห้องปฏิบัติการ และควรปิดฝาภาชนะเก็บฝุ่นให้สนิท (กรมควบคุมมลพิษ, 2547)

อินเทอร์เน็ตของสรรพสิ่ง (Internet of Things)

อินเทอร์เน็ตของสรรพสิ่ง หรือเรียกว่า ไอโอที (IoT) ซึ่งเสนอครั้งแรกโดยเควิน แอชตัน นักเทคโนโลยีชาวอังกฤษ ในปี พ.ศ. 2542 นั้นมีศักยภาพที่จะส่งผลกระทบทุกอย่าง ตั้งแต่โอกาสของผลิตภัณฑ์ใหม่ ๆ ไปจนถึงการเพิ่มประสิทธิภาพพื้นที่ร้านค้า ไปจนถึงการเพิ่มประสิทธิภาพของผู้ปฏิบัติงานโรงงาน ซึ่งจะช่วยเพิ่มพลังให้กับกำไรได้และเป็นที่เชื่อกันว่า IoT นั้นจะช่วยปรับปรุงประสิทธิภาพการใช้พลังงาน การตรวจสอบระยะไกล และการควบคุมสินทรัพย์ทางกายภาพ และประสิทธิภาพการทำงานผ่านแอพพลิเคชั่นที่หลากหลาย เช่น การรักษาความปลอดภัยภายในบ้านไปจนถึงการตรวจสอบสภาพโรงงาน ตอนนี้ IoT ถูกนำมาใช้ในตลาดด้านการดูแลสุขภาพ บ้าน เครื่องใช้ อาคาร ตลาดค้าปลีก บริษัทพลังงานและการผลิต การเคลื่อนย้าย การขนส่ง บริษัทโลจิสติกส์ และตามสื่อต่าง ๆ อุปกรณ์บางอย่างกลายเป็นดิจิทัลมากขึ้นและเชื่อมโยงกันมากขึ้น สร้างเครือข่ายระหว่างเครื่องจักร มนุษย์ และอินเทอร์เน็ต นำไปสู่การสร้างระบบนิเวศใหม่ที่ช่วยให้มีผลผลิตสูงขึ้น ประหยัดพลังงานได้ดีขึ้นและทำกำไรได้สูงขึ้น เซ็นเซอร์ ช่วยในการรับรู้สถานะของสิ่งต่าง ๆ โดยที่พวกเขาได้เปรียบในการคาดคะเน ความต้องการของมนุษย์ตามข้อมูลที่รวบรวมตามบริบท อุปกรณ์อัจฉริยะเหล่านี้ ไม่เพียงแต่รวบรวมข้อมูลจากสภาพแวดล้อมเท่านั้น แต่ยังสามารถตัดสินใจ โดยปราศจากการแทรกแซงของมนุษย์

อุปกรณ์กลายเป็นดิจิทัลมากขึ้นและเชื่อมโยงกันมากขึ้น สร้างเครือข่ายระหว่างเครื่องจักร มนุษย์ และอินเทอร์เน็ต นำไปสู่การสร้างระบบนิเวศใหม่ที่ช่วยให้มีผลผลิตสูงขึ้น ประหยัดพลังงานได้ดีขึ้น และทำกำไรได้สูงขึ้น เซ็นเซอร์ ช่วยในการรับรู้สถานะของสิ่งต่าง ๆ โดยที่พวกเขาได้เปรียบในการคาดคะเนความต้องการของมนุษย์ตามข้อมูลที่รวบรวมตามบริบท อุปกรณ์อัจฉริยะเหล่านี้ไม่เพียงแต่รวบรวมข้อมูลจากสภาพแวดล้อมเท่านั้น แต่ยังสามารถตัดสินใจได้โดยปราศจากการแทรกแซงของมนุษย์ เทคโนโลยี IoT ถูกนำมาใช้ในชีวิตประจำวันของเราสำหรับปลดล็อกประตูโดยไม่มีกุญแจ ในการจดจำการ์ด ล็อคอัตโนมัติ การตรวจจับยานพาหนะ ระบบการชำระค่าผ่านทาง และสำหรับติดตามสัตว์ IoT การบล็อกจะมาจากอุปกรณ์ที่เปิดใช้งานเว็บที่ให้บริการแพลตฟอร์มทั่วไป ซึ่งพวกเขาสามารถสื่อสารและพัฒนาแอพพลิเคชั่นใหม่เพื่อดึงดูดผู้ใช้ (B.K. Tripathy and J. Anuradha, 2018)

ระบบ IoT มีประโยชน์ในการใช้งานที่หลากหลาย

1. ระบบอุตสาหกรรมใช้เซ็นเซอร์เพื่อตรวจสอบทั้งกระบวนการทางอุตสาหกรรม คุณภาพของผลิตภัณฑ์ และสถานะของอุปกรณ์ เช่น เซนเซอร์ที่รวบรวมข้อมูลที่ใช้ทำนายความล้มเหลวของมอเตอร์ที่กำลังจะเกิดขึ้น

2. อาคารอัจฉริยะใช้เซ็นเซอร์เพื่อระบุตำแหน่งของผู้คนและอาคาร ข้อมูลดังกล่าวสามารถใช้ในการควบคุมระบบทำความร้อน ระบายอากาศ ปรับอากาศ และระบบแสงสว่างเพื่อลดต้นทุนการดำเนินงาน ในอาคารระบบอัจฉริยะด้านโครงสร้างยังใช้เซ็นเซอร์เพื่อตรวจสอบความสมบูรณ์ของโครงสร้าง

3. เมืองอัจฉริยะใช้เซ็นเซอร์เพื่อตรวจสอบการจราจรของคนเดินถนนและยานพาหนะ และอาจรวมข้อมูลจากอาคารอัจฉริยะ (Dimitrios Serpanos, & Marilyn Wolf, 2018)

งานวิจัยที่เกี่ยวข้อง

การศึกษาปริมาณฝุ่นตกภายนอกอาคารบริเวณเขตเมืองพิษณุโลก ทำการวางตัวอย่างทั้งหมด 10 จุด โดยแบ่งเป็นบริเวณถนนสายหลักทางแยกที่เป็นเส้นทางการคมนาคม จำนวน 7 จุด และบริเวณย่านการค้าในเขตเทศบาลเมืองพิษณุโลก จำนวน 3 จุด มีลักษณะการวาง 2 แบบ คือ วางในแนวราบ และวางในแนวดิ่ง ทำการเก็บตัวอย่างทุก ๆ 1 วัน เป็นเวลา 1 สัปดาห์ และนำข้อมูลที่ได้มาคำนวณหารปริมาณฝุ่นตกและเปรียบเทียบปริมาณฝุ่นตกในแต่ละพื้นที่ ซึ่งค่าที่ได้ในแต่ละวันจะแตกต่างกันน้อยสุดถึงมากที่สุด จากการเก็บตัวอย่างการทดลองในครั้งนี้ พบว่าบริเวณป้ายรถเมล์วงเวียนรถไฟ มีปริมาณฝุ่นตก 153.76 mg/m2/day ในวันที่ 30 มกราคม 2546 เนื่องจากบริเวณดังกล่าวเป็นเส้นทางคมนาคม เป็นวันที่มีผู้ใช้ถนนทำให้เกิดฝุ่นที่ผิวถนน รวมทั้งการก่อสร้างตลาดสดและการปรับปรุงสถานีรถไฟ และในวันดังกล่าวมีอากาศค่อนข้างร้อนในช่วงบ่ายทำให้มีปริมาณฝุ่นเพิ่มมากขึ้น (ศลีจิตร น้ำจิตร และคณะ, 2545)

สถานที่ก่อสร้างทางโยธาถือเป็นหนึ่งในสภาพแวดล้อมที่เสี่ยงที่สุดซึ่งอาจมีอันตรายเกิดขึ้นมากมาย เพื่อปกป้องคนงานก่อสร้างและป้องกันอุบัติเหตุดังกล่าว บทความนี้มีจุดประสงค์การออกแบบสำหรับระบบอัตโนมัติที่ตรวจสอบ แปลเป็นภาษาท้องถิ่น และเตือนคนงานที่ปฏิบัติงานในเขตอันตราย ซึ่งระบบที่นำเสนอนั้นเป็นมิตรกับผู้ใช้ และการออกแบบนั้นขึ้นอยู่กับอินเทอร์เน็ตของสรรพสิ่ง (IoT) ส่วนประกอบที่แตกต่างกันของการออกแบบนี้ถูกรวมเข้ากับเซิร์ฟเวอร์ออนไลน์แบ็กเอนด์มิดเดิลแวร์อย่างราบรื่น ในการตรวจจับและระบุตัวคนงานก่อสร้างอย่างแม่นยำ การออกแบบอุปกรณ์สวมใส่ประกอบด้วยชุดของส่วนประกอบซึ่งเป็นตัวรับส่งสัญญาณวิทยุ (ตัวส่ง/ตัวรับ) เซ็นเซอร์เตือน ตัวกระตุ้นสัญญาณเตือน และโมดูล GPRS อุปกรณ์สวมใส่นี้จะกลายเป็นแบบไฮบริด (แอคทีฟ/พาสซีฟ) ผ่านระยะใกล้ บทความนี้ยังนำเสนอการใช้งานโหนดไร้สายที่ใช้พลังงานจากพลังงานแสงโดยใช้โฟโตโวลตาเซลล์ โหนดเหล่านี้ใช้แผนการจัดการพลังงานและการจัดเก็บสำหรับการทำงานต่อเนื่องสำหรับสภาพแวดล้อมในร่มและกลางแจ้ง (Riad Kanan et al., 2018)

การศึกษาปริมาณฝุ่นจากโรงโม่หิน บริษัท โชคอนันต์ก่อสร้างอุดรธานี จำกัด อำเภอนากลาง จังหวัดหนองบัวลำภู ในการศึกษาระยะห่างของฝุ่นละอองขนาดเล็กจากปากหม้อถึงชุมชนได้

เก็บตัวอย่างฝุ่นละอองจากสถานีตรวจวัดอากาศ 6 สถานี ผลการศึกษาพบว่าปริมาณฝุ่นละอองในบรรยากาศที่มีขนาดเส้นผ่านศูนย์กลางเล็กกว่า 10 ไมโครเมตร บริเวณปากปล่องที่ไม่เป็นแหล่งมลพิษมีความเข้มข้นของฝุ่นละออง 8.58 ไมโครกรัมต่อลูกบาศก์เมตร แหล่งกำเนิดมลพิษคือบ้านของเจ้าของ สถานประกอบกิจการ นาข้าว ถนนโรงสี และหมู่บ้านนากุดผึ้ง มีความเข้มข้น 0.27 0.13 0.46 และ 0.38 ไมโครกรัมต่อลูกบาศก์เมตร ตามลำดับ ความเข้มข้นของฝุ่นละอองอยู่ที่ 0.23 ไมโครกรัมต่อลูกบาศก์เมตร การวัดความเข้มข้นของฝุ่นพบว่าการฟุ้งกระจายของฝุ่นลดลงตามระยะทางที่เพิ่มขึ้น โดยสรุปการฟุ้งกระจายของฝุ่นละอองขนาดเล็กโรงโม่หิน อำเภอนากลาง จังหวัดหนองบัวลำภู มีความเข้มข้นของฝุ่นละอองขนาดเล็กในบรรยากาศ ค่าจะลดลงตามระยะทางที่เพิ่มขึ้น แสดงให้เห็นว่าที่แหล่งตรวจวัดที่ปากหม้อจนถึงทุ่งนา ส่วนบริเวณถนนโรงโม่หินและหมู่บ้านนากุดผึ้งมีความเข้มข้นสูงกว่าและมีสีแป้งเป็นสีแดง ซึ่งแตกต่างจากปากโรงสีตรงที่ มีฝุ่นสีเทา แสดงว่าจุดดังกล่าวมีมลพิษจากแหล่งอื่นมากกว่า ในการศึกษาครั้งนี้ได้แสดงให้เห็นระยะทางที่สำคัญ คือ ระยะ 1,000 เมตร จากปากโรงสี นาข้าวจะไม่ได้รับผลกระทบ ผลกระทบของฝุ่นต่อการแพร่กระจายจากปากโรงสี กล่าวคือ ฝุ่นละอองขนาดเล็กกว่า 10 ไมโครเมตรสามารถฟุ้งกระจายจากแหล่งกำเนิดปากโรงสีไปได้ไกลถึง 150 เมตร ฝุ่นละอองขนาดเล็กกว่า 10 ไมโครเมตร จะอยู่ในบริเวณโรงสีเท่านั้นและไม่ส่งผลกระทบต่อแหล่งกำเนิดมลพิษภายนอกโรงสี หรือชุมชนใกล้เคียง การศึกษานี้ ผู้วิจัยได้เสนอวิธีการควบคุมปัญหาสิ่งแวดล้อมในโรงโม่หิน และการป้องกันปัญหาเกี่ยวกับสุขภาพ เช่น มาตรการการใช้อุปกรณ์ป้องกันภัยส่วนบุคคล สร้างความตระหนักในการรักษาสุขภาพของคนงาน (นริศรา ไทยนาม, 2552)

การตรวจวัดฝุ่นละอองใช้เครื่องมือตรวจวัดเซ็นเซอร์ (Sensor) และการควบคุมปริมาณฝุ่นโดยใช้ระบบ รดน้ำแบบหมอก (Spray Fogging) การทดลองการวัดปริมาณฝุ่นในโครงการก่อสร้างดำเนินการทดลองในพื้นที่ โครงการก่อสร้าง กองบัญชาการตำรวจสอบสวนกลาง ปากเกร็ด นนทบุรี พื้นที่การทดลองอาคารเก็บของกลาง ขนาด 6 ชั้น โดยทดสอบที่ชั้น 3 ของอาคาร พื้นที่โดยรวมประมาณ 400 ตารางเมตร พบว่าการใช้อุปกรณ์เซ็นเซอร์ (sensor) วัดฝุ่นและการสร้างเครือข่าย IoT ขึ้นมาสำหรับการทดลองวัดค่าปริมาณฝุ่นและควบคุมปริมาณฝุ่นนั้น ผลการทดลองค่อนข้างเป็นที่น่าพอใจ กล่าวคือ อุปกรณ์ตรวจวัดฝุ่นแบบเซนเซอร์สามารถตรวจวัดค่าฝุ่นละอองและส่งข้อมูลไปยังแอปพลิเคชันผ่านการแจ้งเตือนใน LINE Application เพื่อทราบปริมาณฝุ่นละอองแบบเรียลไทม์ (Real-Time) โดยใช้เครือข่ายในระบบ 4G สามารถส่งสัญญาณและทำงานได้ในระยะ 30 - 40 เมตร (จากจุดสัญญาณถึงเซ็นเซอร์) และในระยะ 1 - 40 เมตร ส่งข้อมูลได้อย่างราบรื่น อยู่ในสภาพดี และแบตเตอรี่ที่ใช้เป็นแบตเตอรี่ลิเธียมโพลิเมอร์ขนาด 3.7V 350 mAh ซึ่งต้องใช้ Step Up Converter Power Supply Board 2V/24V To 5V เพื่อแปลงไฟเป็น 5V เพื่อจ่ายไฟให้ทำงานร่วมกับ Node MCU/ESP8266 กำลังไฟ <100 mAh และเซ็นเซอร์ตรวจจับฝุ่น PMS5003 ทำงานกินไฟ <100

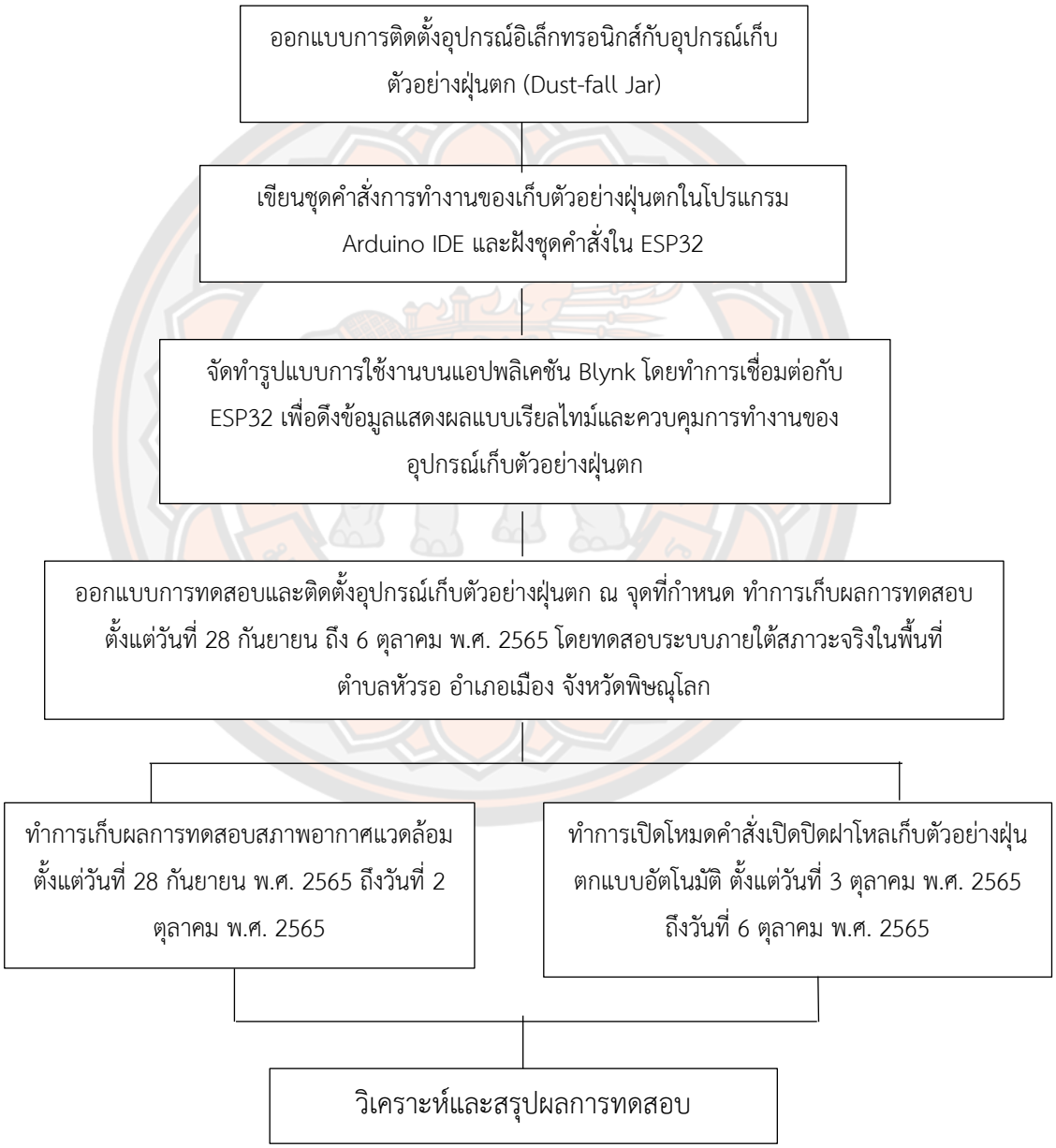
mAh ซึ่งแบตเตอรี่มีประจุไฟ 350 mAh ต้องจ่ายไฟสำหรับการใช้งาน 1.5 ชั่วโมง แต่จากการทดลองพบว่าการใช้งานจริงจะอยู่ที่ประมาณ 2 - 2.30 ชั่วโมง เนื่องจากระดับอินเตอร์เฟส แรงดันเอาท์พุตต่ำสุด (VOL) <0.4V จึงทำให้สิ้นเปลืองไฟฟ้า ใช้งานได้นานขึ้นและสามารถควบคุมคำสั่งเปิด/ปิดได้ ปั๊มน้ำสำหรับควบคุมฝุ่นผ่านแอพพลิเคชั่นทันทีในกรณีที่ฝุ่นเกินมาตรฐาน และการควบคุมปริมาณฝุ่นด้วยการพ่นละอองน้ำสามารถช่วยลดปริมาณฝุ่นจากการทำงานได้ในระดับที่น่าพอใจ (อดิศักดิ์ ประพันธ์กุลและอภินิติ โชติสังกาศ, 2563)

จากศึกษาและทดลองระบบควบคุมการจ่ายน้ำอัจฉริยะด้วยเทคโนโลยี IoT (Intelligent Water Storage and Provision Control using IoT) ระบบสามารถส่งน้ำได้ทั้ง 2 ทาง คือ ควบคุมจากระยะไกล คือ ควบคุมจากสัญญาณอินเทอร์เน็ต คือ เราสามารถสั่งงานเปิด-ปิดปั๊มน้ำผ่าน Dashboard ที่เราออกแบบไว้ตามที่เราต้องการและระยะใกล้ คอนโทรลหรือคอนโทรลหน้างานที่ใช้สวิตช์ปุ่มกดควบคุมการทำงาน เปิด-ปิดน้ำประปา และแผงควบคุมระยะไกลของเรายังสามารถดูสถานะของถังเก็บน้ำว่าอยู่ในระดับสูงหรือต่ำและยังสามารถดูอัตราการไหลของน้ำว่าปั๊มน้ำทำงานหรือไม่จากการไหลของน้ำ เซ็นเซอร์ ค่าจากเซนเซอร์นี้เป็นค่าดิจิตอลแล้ว ก็จะเข้าสู่ Nodemcu V1.0 และทำการอัปโหลดไปยังเว็บไซต์ที่เป็นหน้าควบคุมของเรา จากที่กล่าวมาสรุปได้ว่าระบบสามารถทำงานได้ตามที่เราออกแบบไว้ แต่ก็มีข้อเสียอยู่ส่วนหนึ่ง ของความเสถียรของ Nodemcu V1.0 ในการใช้งาน ซึ่งมีความสำคัญต่อการใช้งานจริง (สิทธิพันธ์ เสือชม และคณะ, 2558)

บทที่ 3

วิธีดำเนินงานวิจัย

แนวทางในการดำเนินงานวิจัย



ภาพ 1 กรอบแนวความคิดการดำเนินงานวิจัย

เครื่องมือที่ใช้ในการวิจัย

1. อุปกรณ์ที่ใช้ในการประยุกต์ใช้งานร่วมกับอุปกรณ์เก็บตัวอย่างฝุ่นตก (Dust-fall Jar) แบบอัจฉริยะ ส่วนของฮาร์ดแวร์

1.1 ESP32-WROOM-32 (ESP32) (ภาพ 2)

1.1.1 ESP32 คือ ไมโครคอนโทรลเลอร์ที่รองรับการเชื่อมต่อ Wi-Fi และ Bluetooth ผลิตโดยบริษัท Espressif ประเทศจีน พัฒนาจากการแก้ไขจุดด้อยต่าง ๆ ของ ESP8266 ปรับให้มีสเปกของฮาร์ดแวร์และเสถียรภาพสูงขึ้น ทั้งยังแก้ไขเรื่อง I/O และ Analog input ที่ไม่เพียงพอต่อการใช้งาน

1.1.2 ESP32 สามารถนำไปใช้พัฒนาด้วยซอฟต์แวร์ เช่น LuaNode, MicroPhython-ESP32, Espruion on ESP32 และ Arduino IDE เป็นต้น โดยส่วนใหญ่ชุดซอฟต์แวร์ที่นิยมใช้พัฒนา คือ Arduino IDE เนื่องจากใช้งานง่ายมีฟังก์ชันต่าง ๆ มากมาย และ ใช้พื้นฐานภาษา C เหมาะสำหรับบุคคลที่เริ่มเรียนรู้การเขียนโปรแกรม

ภาพ 2 ESP32-WROOM-32

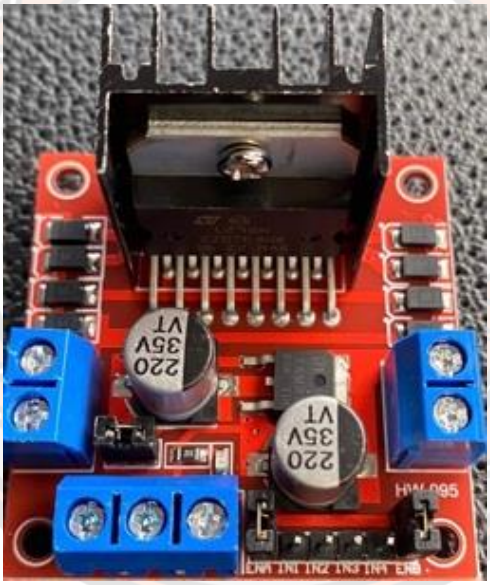
1.2 โมดูลขับมอเตอร์ (ภาพ 3)

L298N คือ ชุดขับมอเตอร์ชนิด H-Bridge ถูกนำมาใช้เพื่อควบคุมทิศทางและความเร็วของมอเตอร์ L298N จะทำงานเมื่อจ่ายกระแสไฟฟ้าให้กับวงจร H-Bridge จากนั้นกระแสไฟฟ้าจะถูกจ่ายให้กับมอเตอร์ตามขั้วที่กำหนดด้วยลอจิกเพื่อควบคุมทิศทาง และความเร็วของมอเตอร์ถูกควบคุมด้วยสัญญาณ PWM (Pulse Width Modulation) ที่มีการปรับความถี่ให้เหมาะสมกับมอเตอร์ที่ใช้งาน

คุณสมบัติของโมดูลขับมอเตอร์ L298N

1. แรงดันสัญญาณลอจิก 5V Drive voltage: 5V - 35V
2. กระแสสัญญาณลอจิก 0 – 36mA
3. กระแสขับมอเตอร์สูงสุด 2A
4. กำลังไฟฟ้าสูงสุด 25W
5. ขนาด 43 x 43 x 26 mm
6. น้ำหนัก 26g

การต่อใช้งานร่วมกับ Arduino ต้องต่อพอร์ต Digital ที่รองรับ PWM เนื่องจากมอเตอร์ใช้สัญญาณ PWM ในการควบคุมความเร็ว การต่อใช้งานสามารถใช้ได้กับขา IN1, IN2, IN3 และ IN4



ภาพ 3 โมดูลขับมอเตอร์ L298N

1.3 เซ็นเซอร์น้ำฝน ความชื้น (ภาพ 4)

โมดูลวัดความชื้นในอากาศและน้ำฝน ค่าที่ได้ออกมาเป็นความต้านทาง (ADC) เมื่ออยู่ในสภาพปกติจะมีความต้านทานสูง หากมีความชื้นมากหรือฝนตกกระทบบริเวณเซ็นเซอร์ในปริมาณมาก ค่าความต้านทานจะลดลง สามารถปรับค่าความไวในการตรวจวัดได้

คุณสมบัติของเซ็นเซอร์

1. แรงดันไฟฟ้า 5V
2. ไฟแสดงสถานะ, LED แสดงสัญญาณเอาต์พุต

3. เอาต์พุตระดับ TTL, สัญญาณเอาต์พุต TTL สำหรับความจุไดรฟ์ระดับต่ำประมาณ 100 MA, สามารถขับรีเลย์ได้

4. การปรับความไวแสดงผ่านโพเทนซิออมิเตอร์

5. ขนาดแผงควบคุม 3 x 1.6 mm

6. ขนาดพื้นที่รับฝน 5.4 x 4.0 mm

ภาพ 4 เซ็นเซอร์น้ำฝน ความชื้น

1.4 โมดูลดิจิตอลเซ็นเซอร์วัดกระแส แรงดัน กำลังไฟฟ้ากระแสตรง (INA219 Digital DC Current Voltage Power Sensor I2C interface Module) (ภาพ 5)

โมดูล INA219 Digital DC Current Voltage Power Sensor ใช้ชิพ INA21 สามารถวัดได้ทั้งแรงดัน กระแส รวมถึงพลังงาน ผ่าน Interface แบบ I2C ความแม่นยำสูง

คุณสมบัติของโมดูล

1. เชื่อมต่อกับไมโครคอนโทรลเลอร์ทาง I2C

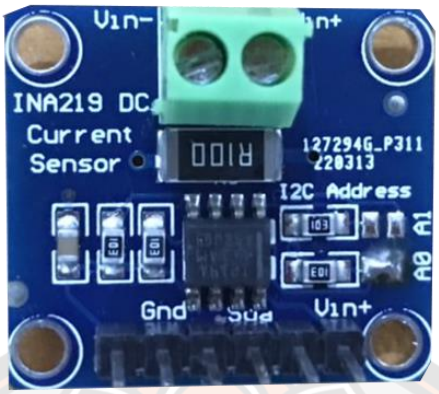
2. ใช้ตรวจจับได้เฉพาะไฟฟ้ากระแสตรงเท่านั้น (ไม่สามารถตรวจจับไฟฟ้ากระแสสลับได้)

3. สามารถตรวจวัดได้ทั้ง 3 อย่างคือ กระแส แรงดัน กำลังไฟฟ้า

4. ไฟเลี้ยง : 3 - 5 VDC

5. แรงดันที่สามารถตรวจจับได้ : 0 - 26 VDC

6. กระแสสูงสุดที่สามารถตรวจจับได้ 3.2 A



ภาพ 5 โมดูลดิจิตอลเซ็นเซอร์วัดกระแส แรงดัน กำลังไฟฟ้ากระแสตรง

1.5 เซ็นเซอร์วัดความเข้มแสง อุณหภูมิ และความชื้น กลางแจ้ง (Light, Temperature, Humidity outdoor transmitter RS485 output) (ภาพ 6)

ชุดเซ็นเซอร์วัดความเข้มแสง อุณหภูมิ และความชื้น แบบกลางแจ้ง มี Modbus RS485 Output ตั้ง Address 1 – 247

พารามิเตอร์ทางเทคนิค

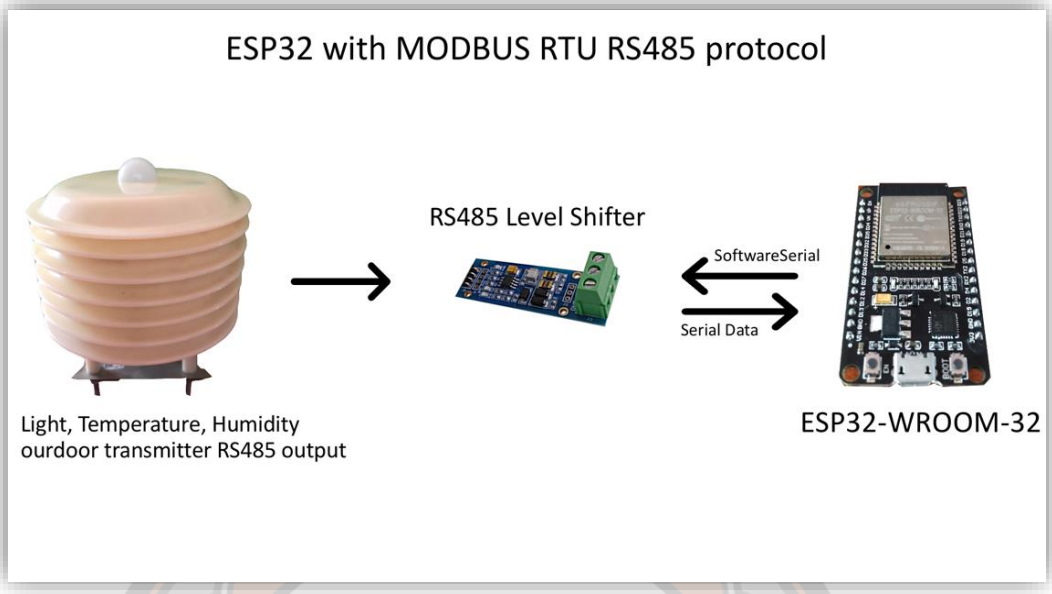
1. แหล่งจ่ายไฟ : 12-24VDC
2. กำลังไฟ : ≤ 0.4W
3. ช่วงการวัดความเข้มของแสง : 0-20000 Lux
4. ช่วงความชื้น : 0%RH-100%RH
5. ช่วงอุณหภูมิ : -40℃ - 80℃
6. ความแม่นยำของความเข้มแสง : ±5%(25℃)
7. ความมั่นคงในระยะยาว : ≤ 5%/y
8. ช่วงแรงดันใช้งาน : ≤ 0.15W(@12V DC , 25℃)
9. สัญญาณเอาต์พุต : RS485
10. การใช้พลังงาน : ≤ 0.15W (@12V DC , 25℃)

ภาพ 6 ชุดเซ็นเซอร์วัดความเข้มแสง อุณหภูมิ และความชื้น แบบกลางแจ้ง

1.6 โมดูล MAX485 TTL to RS485 (ภาพ 7 และภาพ 8)

โมดูล MAX485 TTL to RS485 เป็นตัวแปลงที่ช่วยให้การส่งและรับข้อมูลสามรถทำได้ โดยใช้เครือข่าย RS485 จากไมโครคอนโทรลเลอร์ ช่วยให้การถ่ายโอนข้อมูลระหว่างไมโครคอนโทรลเลอร์และอุปกรณ์ ซึ่งสามารถถ่ายโอนข้อมูลระหว่างอุปกรณ์สูงสุด 32 เครื่อง ผ่านสายข้อมูลเดียวกันด้วยความยาวสายเคเบิลสูงสุด 1.2 km ด้วยอัตราข้อมูลสูงสุด 10 Mbit/s ตัวแปลงนี้ออกแบบมาสำหรับการใช้งานในสำนักงานหรืออุตสาหกรรม (Microdigisoft, 2564)

ภาพ 7 โมดูล MAX485 TTL to RS485



ภาพ 8 การอ่านข้อมูลผ่านพอร์ต Software Serial โดยใช้โมดูล MAX485 TTL to RS485

ที่มา: Microdigisoft, 2564

1.7 มอเตอร์เกียร์ (Gear Motor) (ภาพ 9)

มอเตอร์เกียร์ เป็นอุปกรณ์ไฟฟ้าที่อาศัยหลักการทำงานจากมอเตอร์ในการแปลงพลังงานไฟฟ้าเป็นพลังงานกลทำให้วัตถุเคลื่อนที่ มีฟันเฟืองหรือเกียร์ทำหน้าที่ลดรอบความเร็ว ทดรอบ และเพิ่มแรงบิด โดยที่ระบบเกียร์จะทำหน้าที่ในการลดกำลังของมอเตอร์และลดความเร็วของมอเตอร์ เพราะกำลังและความเร็วของมอเตอร์โดยตรงนั้นมีความเร็วมากเกินไป ไม่เหมาะกับลักษณะงานบางประเภท เช่น งานประเภทสายพาน หรือเป็นการลำเลียง ที่ต้องอาศัยการเคลื่อนไหวไปอย่างช้า ๆ เพราะเหตุนี้จึงต้องอาศัยมอเตอร์เกียร์เข้ามาเป็นตัวช่วยในการทอนกำลังของมอเตอร์ลง ชนิดของมอเตอร์ที่ใช้ในการทดเกียร์นั้น สามารถที่ปรับความช้า หรือว่าความเร็วตามที่ต้องการได้ ยกตัวอย่างเช่น 1/10,1/20,1/30,1/40 เป็นต้น เพื่อปรับให้เหมาะสมกับงานที่ต้องการนำมอเตอร์เกียร์ไปใช้งาน (Jerry Sartara, 2562)

ภาพ 9 มอเตอร์เกียร์ 12VDC

1.8 4G LTE Modem Wi-Fi (ภาพ 10)

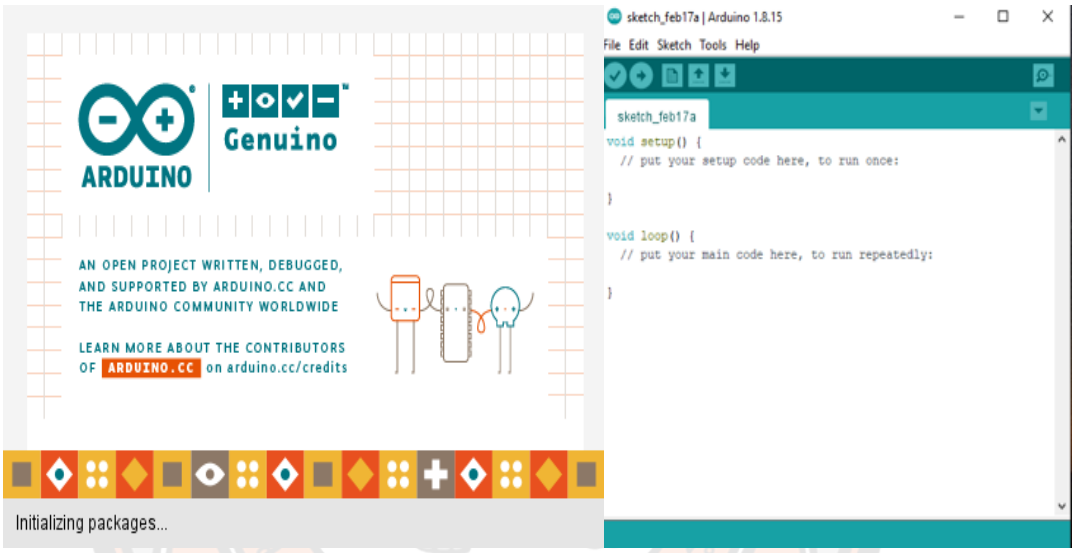
โมเด็มปล่อยสัญญาณ Wi-Fi ใช้ในเชื่อมต่ออุปกรณ์กับสัญญาณ Wi-Fi ในพื้นที่ที่ไม่มีสัญญาณอินเทอร์เน็ต โดยโมเด็มจะใช้งานได้เมื่อต่อเข้ากับไฟฟ้า 5VAC และต้องมีซิมการ์ดที่มีแพคเกตอินเทอร์เน็ตเพื่อปล่อยสัญญาณ

ภาพ 10 โมเด็มปล่อยสัญญาณ Wi-Fi

2. อุปกรณ์ที่ใช้ในการประยุกต์ใช้งานร่วมกับอุปกรณ์เก็บตัวอย่างฝุ่นตก (Dust-fall Jar) แบบอัจฉริยะ ส่วนของซอฟต์แวร์

2.1 โปรแกรม Arduino IDE (ภาพ 11)

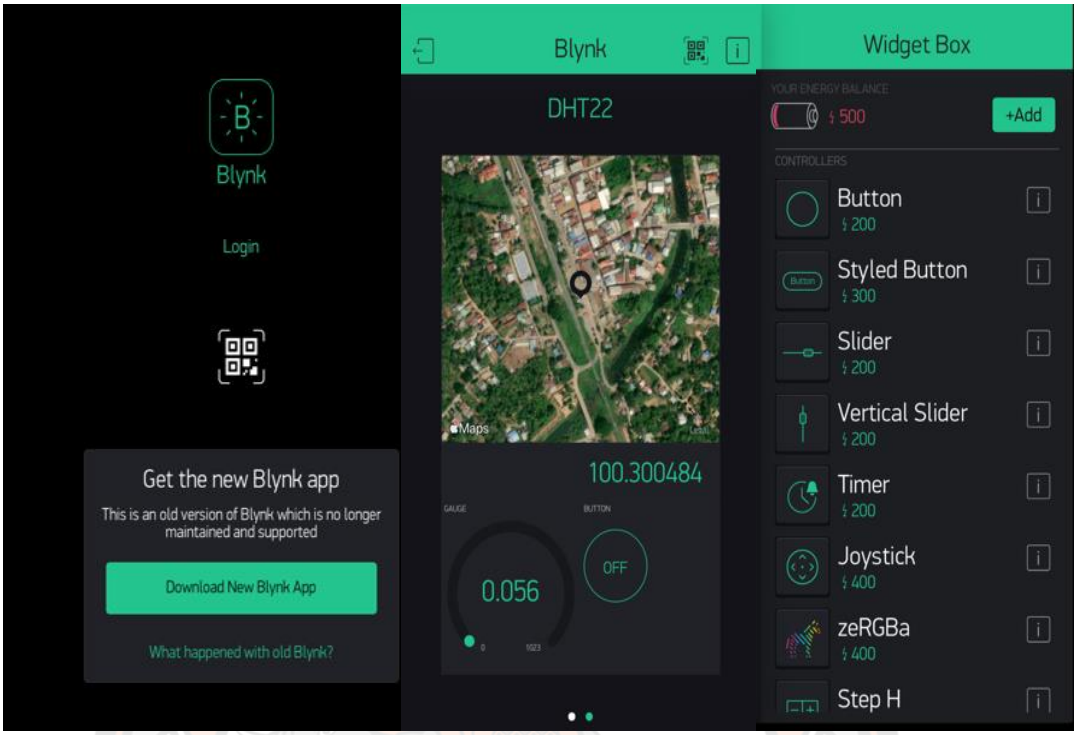
Arduino IDE คือ โปรแกรมที่ใช้สำหรับเขียนโปรแกรม คอมไพล์ และอัปโหลดโปรแกรมลงบอร์ด Arduino หรือบอร์ดอื่น ๆ เช่น NodeMCU ESP8266 และ ESP32 เป็นต้น ภาษาที่ใช้ในการเขียนโปรแกรมส่วนใหญ่ที่นิยมใช้ คือ ภาษา C/C++ สำหรับ Arduino (PoundXI, 2561)



ภาพ 11 การเรียกใช้งานโปรแกรม Arduino IDE

2.2 แอปพลิเคชัน Blynk (ภาพ 12)

Blynk คือ ชุดซอฟต์แวร์ที่ช่วยในการสร้างต้นแบบ ปรับใช้ และจัดการอุปกรณ์อิเล็กทรอนิกส์ที่เชื่อมต่อจากระยะไกล Blynk ออกแบบมาให้ผู้ใช้งานสามมารถเชื่อมต่อฮาร์ดแวร์กับระบบคลาวด์ และสร้างแอปพลิเคชันบน iOS, Android และเว็บไซต์แบบไม่มีโค้ด เพื่อวิเคราะห์ข้อมูลแบบเรียลไทม์หรือเรียกย้อนหลังที่ส่งมาจากอุปกรณ์ นอกจากนั้นยังควบคุมจากระยะไกล รับการแจ้งเตือน และอื่น ๆ อีกมากมาย มีโซลูชันท์เลเบล (ส่วนหนึ่งของแผนธุรกิจ) หมายความว่าผู้ใช้งานสามารถเพิ่มโลโก้บริษัท ไอคอนแอป เลือกธีม สี และเผยแพร่แอปพลิเคชันไปยัง App Store และ Google Play ได้ (Blynk, 2564)



ภาพ 12 การเรียกใช้งานแอปพลิเคชัน Blynk และตัวเลือกการใช้งานต่าง ๆ

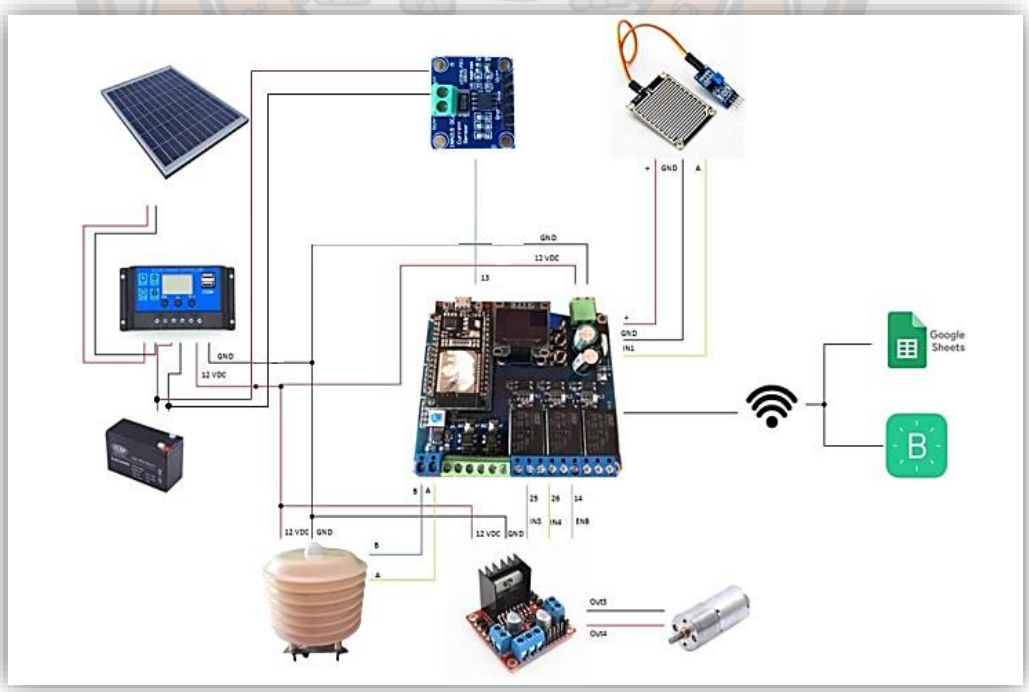
2.3 อุปกรณ์เก็บตัวอย่างฝุ่นตก (Dust fall Jar) แบบอัจฉริยะ (ภาพ 13)

อุปกรณ์เก็บตัวอย่างฝุ่นตก (Dust fall Jar) แบบอัจฉริยะ เป็นอุปกรณ์เก็บตัวอย่างฝุ่นตกในบรรยากาศ ซึ่งได้ออกแบบให้มีระบบป้องกันฝนตก โดยมีอุปกรณ์อิเล็กทรอนิกส์ที่สามารถทำงานได้อัตโนมัติหากมีฝนตก เพื่อลดปัญหาและกระบวนการในการวิเคราะห์ผล อุปกรณ์มีการออกแบบโดยอ้างอิงมาตรฐาน American Society for Testing and Materials (ASTM) รหัส D1739-98 (2017) อุปกรณ์มีความสูงจากพื้นถึงปากโหลเก็บตัวอย่างฝุ่นตก 150 เซ็นติเมตร โหลเก็บตัวอย่างฝุ่นตกมีขนาดเส้นผ่านศูนย์กลาง 15 เซ็นติเมตร สูง 30 เซ็นติเมตร และแผ่นบังลมมีขนาดเส้นผ่านศูนย์กลาง 45 เซ็นติเมตร ระบบป้องกันฝนตกอัตโนมัติจะเริ่มทำงานเมื่อเซ็นเซอร์วัดน้ำฝนและความชื้นในอากาศได้รับน้ำฝน ละอองน้ำหรือความชื้น ทำให้ค่าความต้านทานลดลงถึงค่าที่กำหนดไว้ จากนั้นไมโครคอนโทรลเลอร์จะสั่งให้มอเตอร์ทำการเปิดปิด ฝาครอบที่บริเวณปากโหลเก็บตัวอย่างฝุ่นตกเพื่อป้องกันน้ำฝน หลังจากฝนหยุดตกระบบจะกลับมาทำงานปกติเพื่อเก็บตัวอย่างฝุ่นตกโดยอัตโนมัติ

ภาพ 13 อุปกรณ์เก็บตัวอย่างฝุ่นตก (Dust-fall Jar) แบบอัจฉริยะ

3. การออกแบบระบบเซ็นเซอร์และระบบควบคุมอุปกรณ์ระยะไกล

3.1 องค์ประกอบของระบบ



ภาพ 14 ผังการเชื่อมต่อและการทำงานของอุปกรณ์

จากภาพ 14 แสดงกระบวนการทำงานทั้งหมดของระบบเมื่อจ่ายกระแสไฟฟ้าให้กับระบบ โดยบอร์ด ESP32 จะรับข้อมูลจากเซ็นเซอร์ จากนั้นส่งข้อมูลไปยังอุปกรณ์ที่ได้เขียนคำสั่งไว้ตามเงื่อนไขที่กำหนดดังนี้การทำงานแบบออฟไลน์ เมื่อฝนตกเซ็นเซอร์น้ำฝนมีค่าความต้านทานลดลงจนถึงค่าที่กำหนด ESP32 จะสั่งให้โมดูลขับมอเตอร์ทำงานและจ่ายกระแสไฟฟ้าให้กับมอเตอร์ เพื่อทำการปิดฝาโหลเก็บตัวอย่างฝุ่นตก เมื่อฝนหยุดตกเซ็นเซอร์น้ำฝนจะมีค่าความต้านทานเพิ่มขึ้นจนถึงค่าที่กำหนดไว้ โมดูลขับมอเตอร์ทำงานและจ่ายกระแสไฟฟ้าให้กับมอเตอร์เพื่อทำการเปิดฝาโหลเก็บตัวอย่างฝุ่นตกและเก็บตัวอย่างฝุ่นตกต่อไป

การทำงานแบบออนไลน์ เมื่อ ESP32 ได้รับค่าจากเซ็นเซอร์วัดความเข้มแสงอาทิตย์ อุณหภูมิอากาศ ความชื้นอากาศ เซ็นเซอร์น้ำฝน และโมดูลวัดค่ากระแสไฟฟ้า ข้อมูลจะถูกส่งผ่านเครือข่ายอินเทอร์เน็ตไปยังแอปพลิเคชัน Blynk เพื่อแสดงผลข้อมูลทั้งหมดบนโทรศัพท์มือถือหรือแทบเล็ด และมีฟังก์ชันการส่งคำสั่งเปิดปิดฝาครอบโหลเก็บตัวอย่างฝุ่นตกจากแอปพลิเคชัน Blynk ผ่านเครือข่ายอินเทอร์เน็ตมายัง ESP32 เพื่อสั่งให้โมดูลขับมอเตอร์ทำงานและจ่ายกระแสไฟฟ้าให้กับมอเตอร์เพื่อทำการเปิดปิดฝาโหลเก็บตัวอย่างฝุ่นตก อีกทั้งยังสมารถบันทึกผลข้อมูลต่าง ๆ ส่งไปเก็บไว้ยัง Google Sheets ไว้ใช้วิเคราะห์ผลได้อีกด้วย

เงื่อนไขการทำงานทั้งหมดสามารถทำได้ผ่านการเขียนคำสั่งภาษาซีลงในโปรแกรม Arduino IDE และบันทึกลงในบอร์ด ESP32 โดยใช้พลังงานไฟฟ้าจากระบบเซลล์แสงอาทิตย์แบบ Off Grid เป็นแหล่งจ่ายพลังงานหลักให้แก่ระบบ

3.2 ชุดคำสั่งที่ใช้ในการทำงานของระบบ

การเขียนโปรแกรม Arduino IDE นิยมใช้ภาษาซีในการเขียน โดยรูปแบบของโปรแกรมประกอบด้วยโครงสร้างของฟังก์ชันต่าง ๆ รวมกัน ดังนี้

3.2.1 ชุดคำสั่งส่วนของ Header

Header คือ ส่วนของ Compiler Directive ชุดคำสั่งประกาศค่าคงที่ และตัวแปรต่าง ๆ ที่ใช้ในโปรแกรม (ภาพ 15)

```
#define BLYNK_TEMPLATE_ID "xxxxxxxxxx"
#define BLYNK_DEVICE_NAME "xxxxxxxxxxxx"
//#define BLYNK_TEMPLATE_ID "xxxxxxxxxxx"
//#define BLYNK_DEVICE_NAME "Test1"
#define BLYNK_FIRMWARE_VERSION        "0.1.0"
#define BLYNK_PRINT Serial
//#define BLYNK_DEBUG
#define APP_DEBUG

//#define USE_WROVER_BOARD
//#define USE_TTGO_T7
//#define USE_ESP32C3_DEV_MODULE
//#define USE_ESP32S2_DEV_KIT
#include "BlynkEdgent.h"
#include <SPI.h>
#include <Wire.h>
#include <Adafruit_GFX.h>
```

ภาพ 15 ตัวอย่างชุดคำสั่งส่วนของ Header

3.2.2 ชุดคำสั่งส่วนของ Setup

Setup คือ ส่วนของฟังก์ชันที่กำหนดชุดคำสั่งให้โปรแกรมทำงานเมื่อเริ่มต้นการทำงานเพียงครั้งเดียวของการทำงานโปรแกรมครั้งแรก ฟังก์ชันนี้ใช้สำหรับ Setup ค่าที่ใช้ในการทำงานต่าง ๆ เช่น การกำหนดขาการใช้ของไมโครคอนโทรลเลอร์ (ภาพ 16)

```
/*****************************Setup*****************************/
void setup()
{

 /************************Motor Pin***************************/
 pinMode(motor1Pin1, OUTPUT);
 pinMode(motor1Pin2, OUTPUT);
 pinMode(enable1Pin, OUTPUT);
 // configure LED PWM functionalitites
 ledcSetup(pwmChannel, freq, resolution);
 // attach the channel to the GPIO to be controlled
 ledcAttachPin(enable1Pin, pwmChannel);

 /************************Serial.begin*****************************/
 Serial.begin(9600);
 /*************************************************/
 ina219_1.begin();
 /************************Oled**************************/

 oled1.init();
 oled1.flipScreenVertically();
 oled1.setFont(ArialMT_Plain_10);
```

ภาพ 16 ตัวอย่างชุดคำสั่งส่วนของ Setup

3.2.3 ชุดคำสั่งส่วนของ Loop

Loop คือ ส่วนของฟังก์ชันที่กำหนดชุดคำสั่งให้โปรแกรมทำงานวนซ้ำ ๆ ตามลำดับและเงื่อนไขที่กำหนด (ภาพ 17)

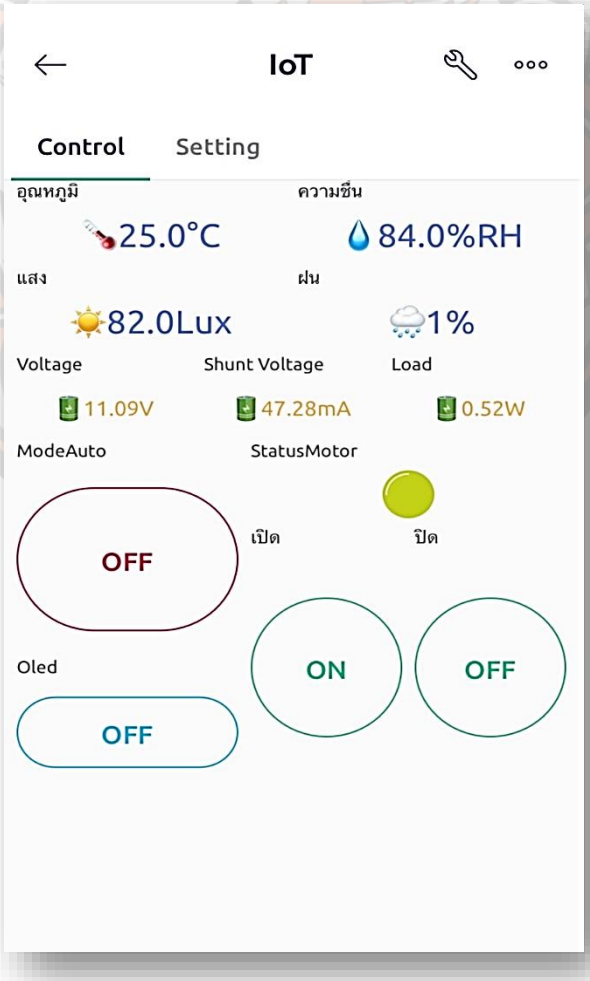
```
/*****************************Loop*****************************/
void loop() {
  BlynkEdgent.run();
  timer.run();
}/************************TH-031 ************************/
int get_modbusRTU_sensor(int addr , uint16_t  REG) {
  int result;
  node.begin(addr, Serial);
  result = node.readInputRegisters (REG, 4); ///< Modbus function 0x04 Read Input Registers
  result = node.readHoldingRegisters (REG, 4);
  if (result == node.ku8MBSuccess) {
    return node.getResponseBuffer(0);
  } else {
    //Serial.print("Connec modbus fail. REG >>> "); mySerial.println(REG, HEX); // Debug
  }
}
/************************TH-031*************************/
void get_LTH_TM() {
  Serial.begin(9600);
  Humidity = get_modbusRTU_sensor(slaveID, HUMIDITY_REG) / 10;
  Temperature = get_modbusRTU_sensor(slaveID, TEMPERATURE_REG) / 10;
  Light = get_modbusRTU_sensor(slaveID, Light_REG);
  Serial.print("\n");
  Serial.print("Temperature = " + String(Temperature, 1) + " C" + "\n");
  Serial.print("Humidity = " + String(Humidity, 1) + " %RH" + "\n");
  Serial.print("Light = " + String(Light, 1) + " lux" + "\n");
  Blynk.virtualWrite(V0, String(Temperature, 1));
  Blynk.virtualWrite(V1, String(Humidity, 1));
   Blynk.virtualWrite(V2, String(Light, 1));
}
```

ภาพ 17 ตัวอย่างชุดคำสั่งส่วนของ Loop

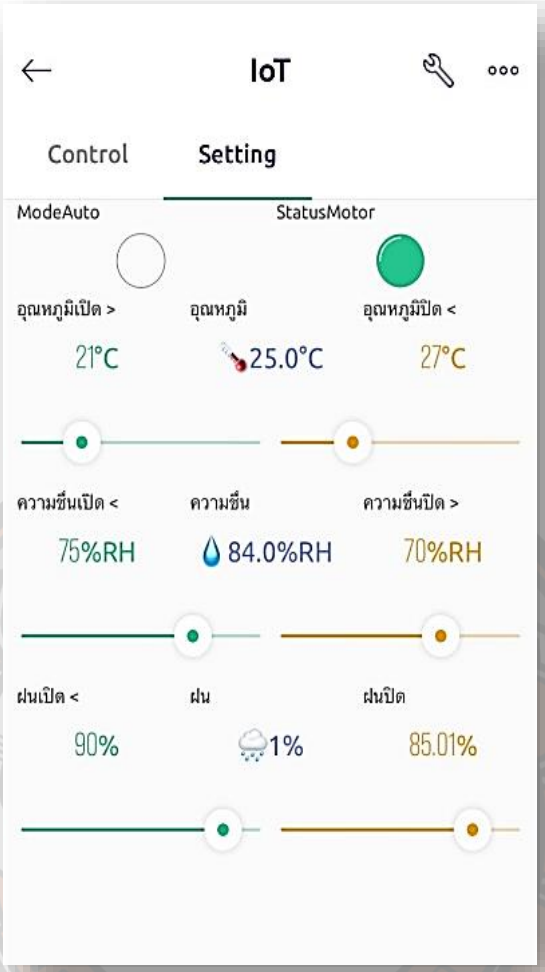
การสร้างรูปแบบการใช้งานหน้าแอปพลิเคชัน Blynk บนสมาร์ทโฟน

การสร้างหน้า Dashboard ของงานวิจัยนี้ ได้กำหนดให้มีการแสดงข้อมูลค่าของอุณหภูมิอากาศ ความชื้นอากาศ ความเข้มแสงอาทิตย์ น้ำฝน และแรงดันไฟฟ้า ที่ส่งมาจากเซ็นเซอร์แบบเรียลไทม์ และกำหนดปุ่มเปิดปิดฝาโหลเก็บตัวอย่างฝุ่นตกโดยจะใช้งานได้ในโหมด manual หรือ auto ซึ่งโหมด auto สามารถกำหนดค่าการทำงานของมอเตอร์ได้ตามต้องการตามสภาพพื้นที่และยังสามารถเปิดดูข้อมูลผ่านจอที่ติดตั้งบนอุปกรณ์ผ่านคำสั่งจากแอปพลิเคชัน (ภาพ 18) บนสมาร์ทโฟนได้อีกด้วย

การปรับค่าการทำงานของมอเตอร์ในโหมด auto (ภาพ 19) เพื่อเพิ่มความแม่นยำในการทำงานของการเปิดปิดฝาโหลเก็บตัวอย่างฝุ่นตก โดยจะปรับค่าของอุณหภูมิอากาศ ความชื้นอากาศ และค่าน้ำฝน ให้อยู่ช่วงที่มีโอกาสเกิดฝนตกได้มากที่สุดโดยอ้างอิงจากข้อมูลที่เก็บได้จากการติดตั้งอุปกรณ์



ภาพ 18 หน้า Dashboard การใช้งานบนสมาร์ทโฟนในแอปพลิเคชัน Blynk

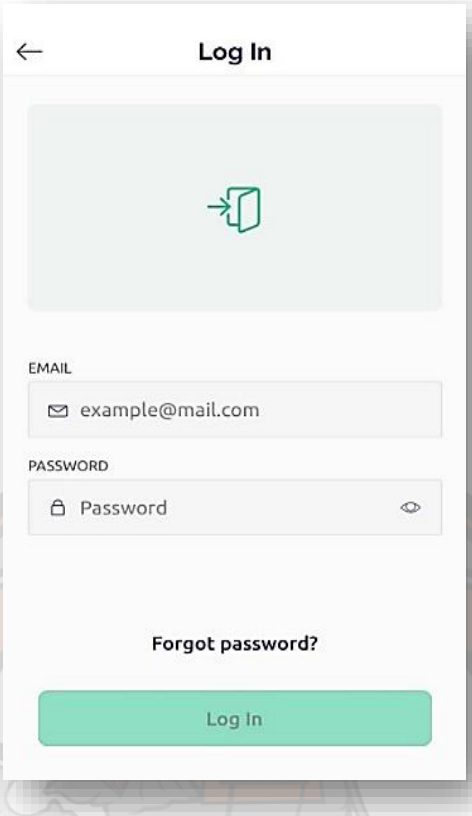


ภาพ 19 หน้า Dashboard การปรับค่าการทำงานของมอเตอร์ในโหมด auto

1. ขั้นตอนการสร้างรูปแบบการใช้งานหน้าแอปพลิเคชัน Blynk

1.1 การลงทะเบียนใช้งานแอปพลิเคชัน Blynk (ภาพ 20)

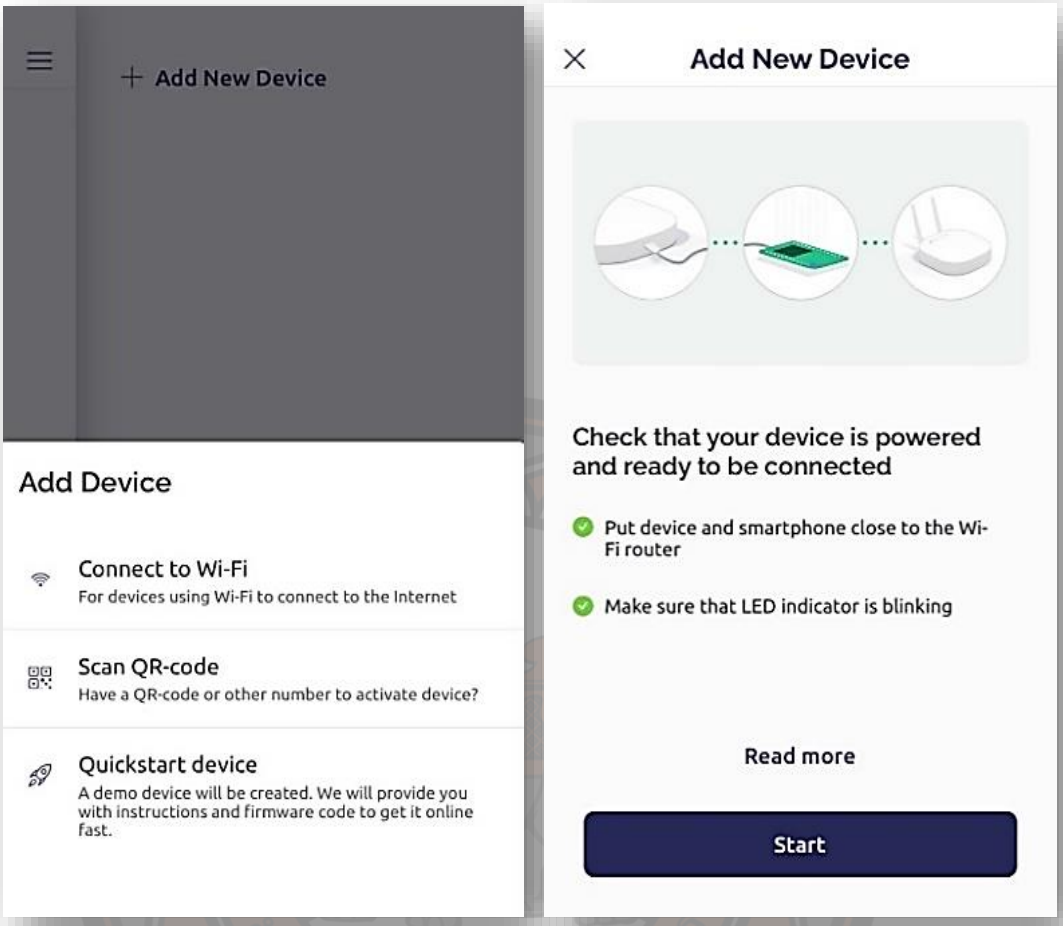
เมื่อติดตั้งแอปพิลเคชันแล้ว ซึ่งใช้งานได้ในเว็บไซต์และสมาร์ทโฟน ทั้งในระบบ android และ ios การเข้าสู่ระบบแอปพลิเคชัน Blynk ต้องสมัครลงทะเบียนผู้ใช้งานโดยใช้อีเมลเพื่อยืนยันตัวตนและตั้งรหัสผ่านการใช้งาน



ภาพ 20 การลงทะเบียนใช้งานแอปพลิเคชัน Blynk

1.2 การเชื่อมต่อสมาร์ทโฟนกับอุปกรณ์ผ่าน Wi-Fi (ภาพ 21)

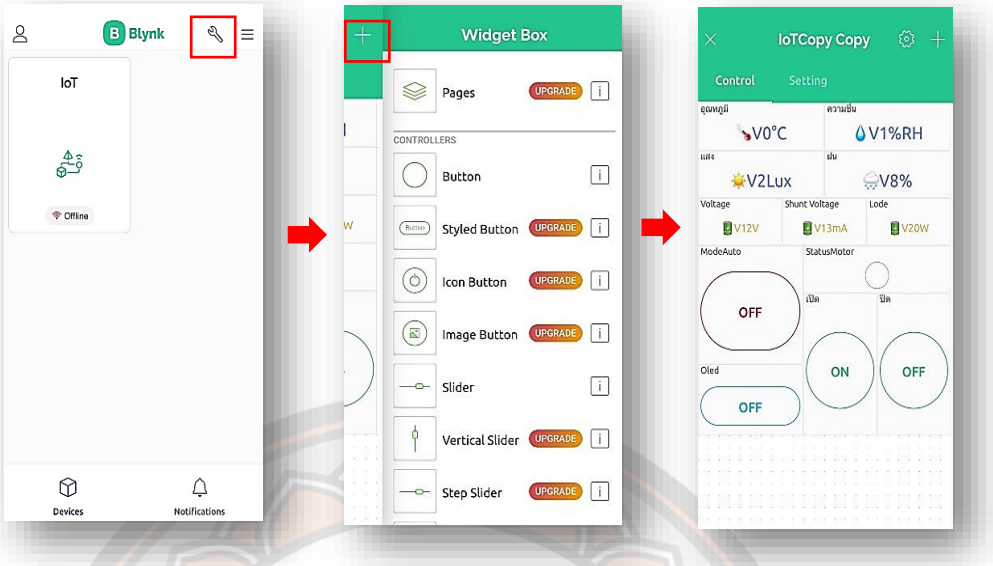
การใช้งานแอปพลิเคชัน Blynk ในสมาร์ทโฟน ต้องทำการเชื่อมต่อกับอุปกรณ์ผ่าน Wi-Fi โดยที่บอร์ด ESP32 ต้องเขียนชุดคำสั่งการเชื่อมต่อ Wi-Fi และรหัส Token ที่ได้จากแอปพลิเคชัน Blynk ไว้ เมื่อเชื่อมต่อเสร็จจะขึ้นหน้า Dashboard ให้เราสามารถปรับแต่งได้ตามการใช้งาน



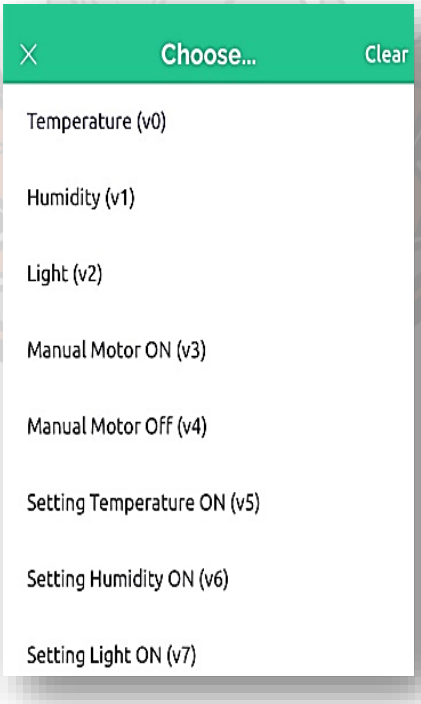
ภาพ 21 การเชื่อมต่อสมาร์ทโฟนกับอุปกรณ์ผ่าน Wi-Fi ในแอปพลิเคชัน Blynk

1.3 การสร้างหน้า Dashboard สำหรับใช้งานบนสมาร์ทโฟน

เมื่อทำการเชื่อมต่ออุปกรณ์เสร็จเรียบร้อย ให้เลือกไปที่รูปประแจหน้าแอปพลิเคชันจากนั้นเลือกไปที่เครื่องหมายบวก จะขึ้น Widget ที่เราสามารถเลือกไปวางบนหน้าแอปพลิเคชันเพื่อจัดรูปแบบการใช้งานได้ตามที่ถูกออกแบบไว้ โดยในการทดสอบนี้ได้ออกแบบให้มี Widget ในส่วนของการแสดงผลข้อมูลของค่าที่ได้จากเซ็นเซอร์ต่าง ๆ มีปุ่มเปิดใช้การทำงานของโหมด manual และ auto ปุ่มเปิดจอ OLED และปุ่มเปิดปิดฝาโหลดเก็บตัวอย่างฝุ่นตกที่ทำงานด้วยมอเตอร์ (ภาพ 22) ข้อมูลที่ Widget แสดงค่าจะถูกนำมาจากชุดคำสั่งที่ถูกกำหนดไว้ในบอร์ด ESP32 ตัวอย่าง เช่น เมื่อเขียนชุดคำสั่งในโปรแกรม Arduino IDE ให้ V0 เป็นค่าจากเซ็นเซอร์อุณหภูมิอากาศ หรือ V3 เป็นค่าจากการสั่งเปิดมอเตอร์แบบ manual เป็นต้น ข้อมูลที่ได้จะแสดงขึ้นบนการตั้งค่า Widget บนแอปพลิเคชัน ซึ่งสามารถเลือกข้อมูลไปใช้ได้ตามรูปแบบที่กำหนด (ภาพ 23)

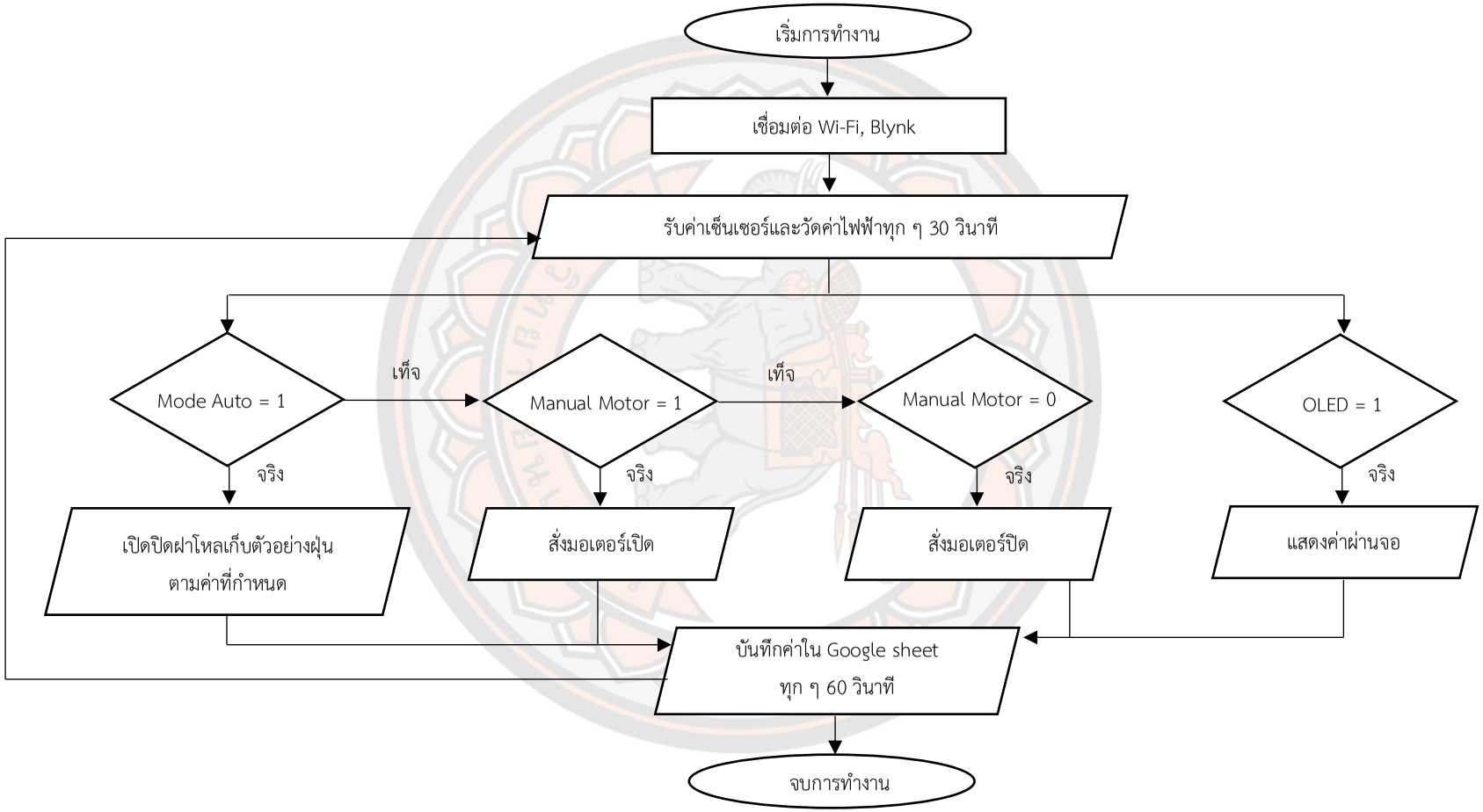


ภาพ 22 การสร้างหน้า Dashboard ในแอปพลิเคชัน Blynk



ภาพ 23 การตั้งค่าการรับข้อมูลในส่วน Widget

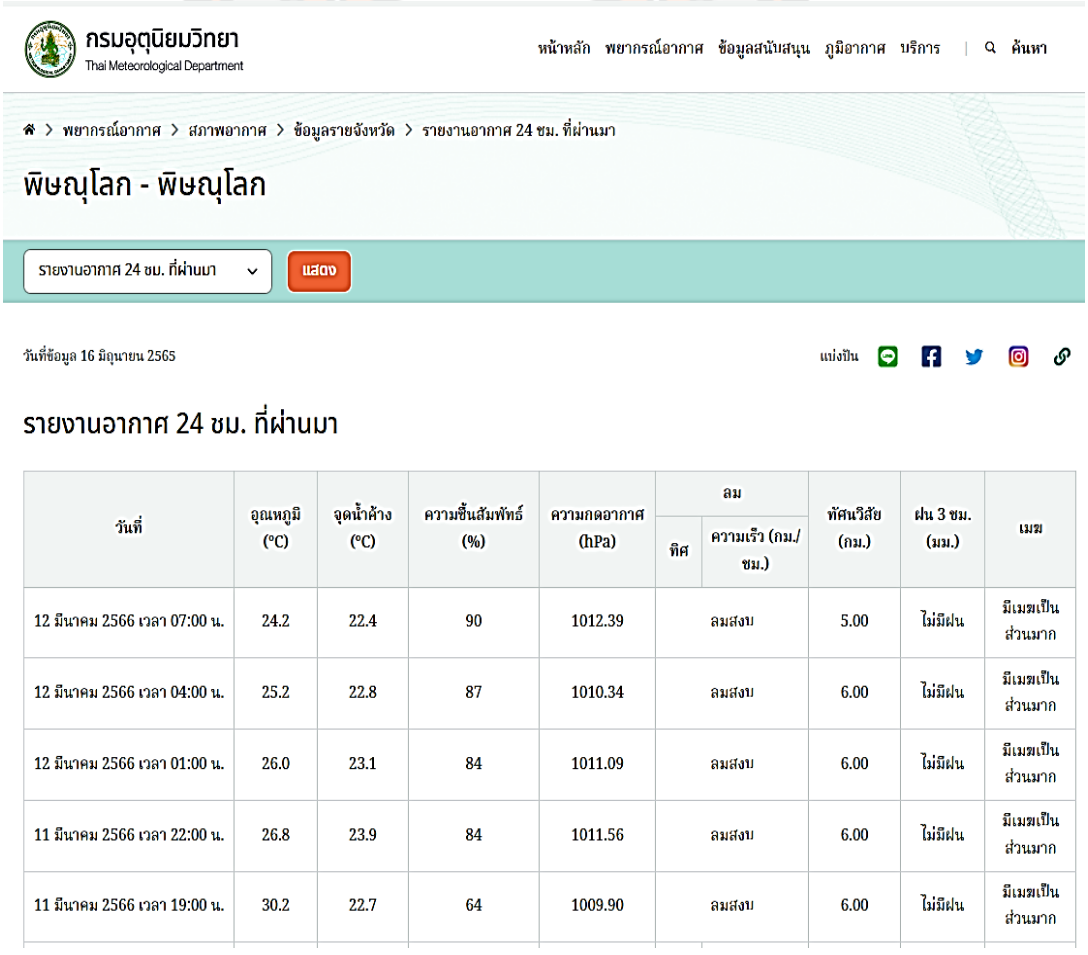
ขั้นตอนการทำงานของระบบ



ภาพ 24 แผนผังขั้นตอนการทำงานของระบบ

การสอบเทียบเซ็นเซอร์วัดความเข้มแสง อุณหภูมิ และความชื้น กลางแจ้ง (Light, Temperature, Humidity outdoor transmitter RS485 output)

การทดสอบการประยุกต์ใช้เทคโนโลยีอินเทอร์เน็ตของสรรพสิ่งร่วมกับอุปกรณ์เก็บตัวอย่างฝุ่นตกแบบอัจฉริยะ จำเป็นต้องตรวจสอบความถูกต้องของเครื่องที่ใช้ในการตรวจวัดและบันทึกผล เพื่อป้องกันความผิดพลาดในการดำเนินการบันทึกผลข้อมูล การตรวจสอบเซนเซอร์ที่ใช้ในการตรวจวัดและบันทึกผลข้อมูล โดยการนำข้อมูลที่บันทึกได้จากเซ็นเซอร์วัดความเข้มแสง อุณหภูมิ และความชื้น กลางแจ้ง (Light, Temperature, Humidity outdoor transmitter RS485 output) มาเปรียบเทียบกับข้อมูลของทางเว็บไซต์กรมอุตุนิยมวิทยาพิษณุโลก ดังภาพ 25 ตั้งแต่เวลา 19.00 น. – 07.00 น. ของวันที่ 11 มีนาคม พ.ศ.2566 ถึงวันที่ 12 มีนาคม พ.ศ.2566 เป็นเวลา 12 ชั่วโมง



กรมอุตุนิยมวิทยา
Thai Meteorological Department

หน้าหลัก พยากรณ์อากาศ ข้อมูลสนับสนุน ภูมิอากาศ บริการ | ค้นหา

พยากรณ์อากาศ > สภาพอากาศ > ข้อมูลรายจังหวัด > รายงานอากาศ 24 ชม. ที่ผ่านมา

พิษณุโลก - พิษณุโลก

รายงานอากาศ 24 ชม. ที่ผ่านมา | แสดง

วันที่ข้อมูล 16 มิถุนายน 2565 แบ่งปัน

รายงานอากาศ 24 ชม. ที่ผ่านมา

วันที่	อุณหภูมิ (°C)	จุดน้ำค้าง (°C)	ความชื้นสัมพัทธ์ (%)	ความกดอากาศ (hPa)	ลม		ทัศนวิสัย (กม.)	ฝน 3 ชม. (มม.)	เมฆ
					ทิศ	ความเร็ว (กม./ชม.)			
12 มีนาคม 2566 เวลา 07:00 น.	24.2	22.4	90	1012.39	ลมสงบ		5.00	ไม่มีฝน	มีเมฆเป็นส่วนมาก
12 มีนาคม 2566 เวลา 04:00 น.	25.2	22.8	87	1010.34	ลมสงบ		6.00	ไม่มีฝน	มีเมฆเป็นส่วนมาก
12 มีนาคม 2566 เวลา 01:00 น.	26.0	23.1	84	1011.09	ลมสงบ		6.00	ไม่มีฝน	มีเมฆเป็นส่วนมาก
11 มีนาคม 2566 เวลา 22:00 น.	26.8	23.9	84	1011.56	ลมสงบ		6.00	ไม่มีฝน	มีเมฆเป็นส่วนมาก
11 มีนาคม 2566 เวลา 19:00 น.	30.2	22.7	64	1009.90	ลมสงบ		6.00	ไม่มีฝน	มีเมฆเป็นส่วนมาก

ภาพ 25 การเปรียบเทียบกับข้อมูลจากเว็บไซต์กรมอุตุนิยมวิทยาพิษณุโลก ตั้งแต่เวลา 19.00 น. – 07.00 น. ของวันที่ 11 มีนาคม พ.ศ.2566 ถึงวันที่ 12 มีนาคม พ.ศ.2566

ตาราง 5 ผลการสอบเทียบเซ็นเซอร์วัดความเข้มแสง อุณหภูมิ และความชื้น กลางแจ้ง (Light, Temperature, Humidity outdoor transmitter RS485 output)

วันที่	เวลา	อุณหภูมิ (องศาเซลเซียส)	อุณหภูมิ กรมอุตุนิยมวิทยา พิษณุโลก (องศาเซลเซียส)	ค่าความคาดเคลื่อน	ความชื้น (%RH)	ความชื้น กรมอุตุนิยมวิทยา พิษณุโลก (%RH)	ค่าความคาดเคลื่อน
11/3/66	19:00	30	30.2	0.7	61	64	4.7
11/3/66	22:00	27	26.8	0.7	80	84	4.8
12/3/66	1:00	26	26	0.0	80	84	4.8
12/3/66	4:00	25	25.2	0.8	82	87	5.7
12/3/66	7:00	24	24.2	0.8	86	90	4.4

การติดตั้งและการเก็บข้อมูลการทดสอบอุปกรณ์เก็บตัวอย่างฝุ่นตก

ภาพ 26 ตำแหน่งการติดตั้งอุปกรณ์เก็บตัวอย่างฝุ่นตก

การติดตั้งอุปกรณ์เก็บตัวอย่างฝุ่นตก ได้ทำการติดตั้ง ณ ตำบลหัวรอ อำเภอเมือง จังหวัดพิษณุโลก โดยทำการติดตั้งกลางแจ้งเพื่อทดสอบกับสภาวะอากาศจริง ตำแหน่งของแผงโซล่าเซลล์หันไปทางทิศใต้ เพื่อใช้ในการผลิตและกักเก็บพลังงานที่ใช้ในการดำเนินการทดสอบตลอด 24 ชั่วโมง การดำเนินการเก็บข้อมูลเริ่มเก็บตั้งแต่วันที่ 28 กันยายน พ.ศ. 2565 ถึงวันที่ 6 ตุลาคม พ.ศ. 2565 บันทึกข้อมูลทุก ๆ 1 นาที เป็นเวลา 24 ชั่วโมง (ภาพ 25) โดยจะแบ่งการเก็บผลการทดสอบเป็น 2 กรณี ได้แก่

1. ทำการเก็บข้อมูลของเซ็นเซอร์ในการวัดค่าอุณหภูมิอากาศ ความชื้นอากาศ และความเข้มแสงอาทิตย์ เก็บข้อมูลในวันที่ 28 กันยายน พ.ศ. 2565 ถึงวันที่ 2 ตุลาคม พ.ศ. 2565

2. ทำการเปิดโหมดคำสั่งเปิดปิดฝาโหลเก็บตัวอย่างฝุ่นตกแบบอัตโนมัติ เก็บข้อมูลในวันที่ 3 ตุลาคม พ.ศ. 2565 ถึงวันที่ 6 ตุลาคม พ.ศ. 2565

จากนั้นนำข้อมูลที่ได้จากการทดสอบโดยการเขียนคำสั่งให้ชุดข้อมูลต่าง ๆ ถูกส่งไปเก็บไว้ยัง Google Sheets เพื่อบันทึกผลของข้อมูลทั้งหมดและนำข้อมูลมาวิเคราะห์และอภิปรายผลการทดสอบอุปกรณ์เก็บตัวอย่างฝุ่นตกต่อไป

บทที่ 4

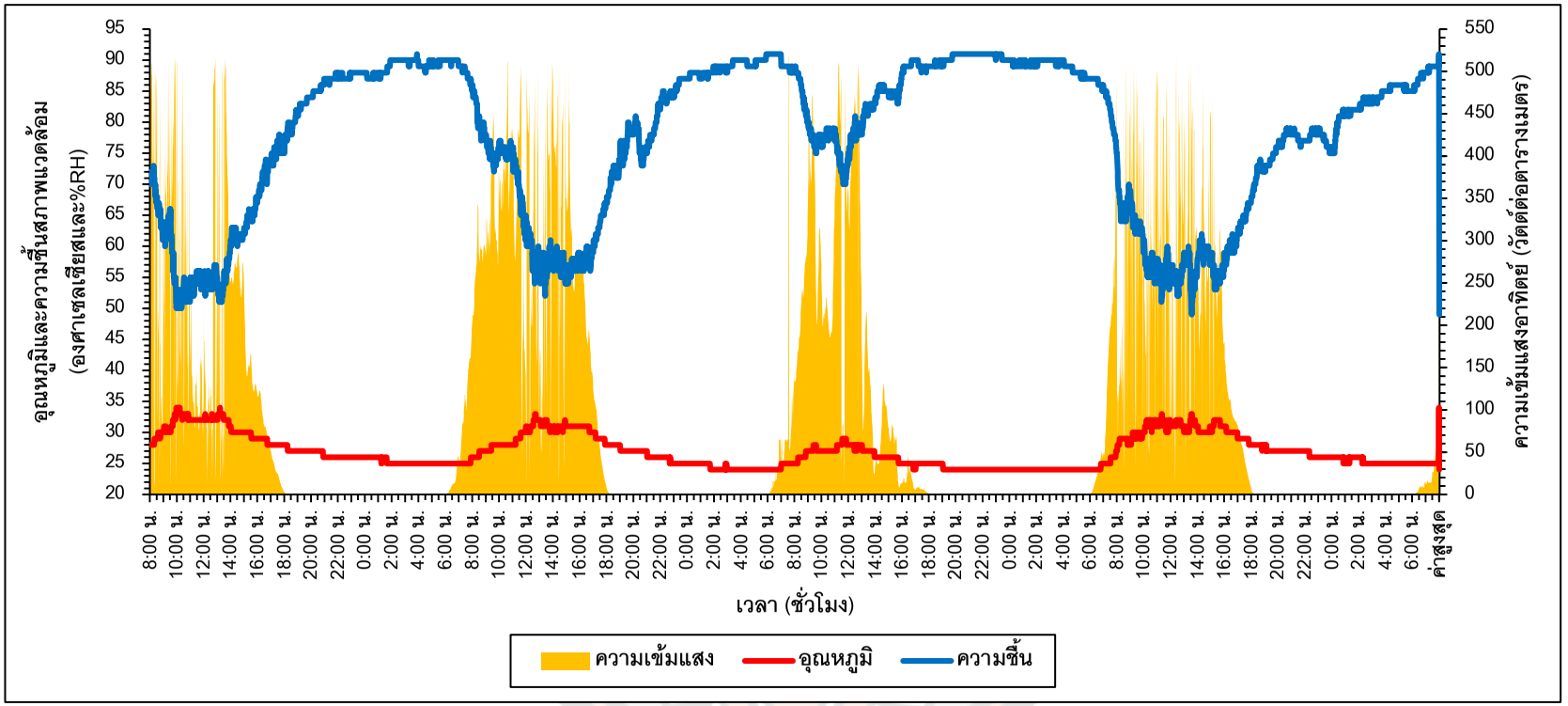
ผลการวิจัย

งานวิจัยนี้เป็นการประยุกต์ใช้เทคโนโลยีอินเทอร์เน็ตของสรรพสิ่งร่วมกับอุปกรณ์เก็บตัวอย่างฝุ่นตกแบบอัจฉริยะ โดยทำการติดตั้งเซ็นเซอร์เพื่อวัดค่าอุณหภูมิอากาศ ความชื้นอากาศ และความเข้มแสงอาทิตย์ บริเวณพื้นที่ทำการทดสอบ และทำการเปิดโหมดเปิดปิดฝาโหลเก็บตัวอย่างฝุ่นตกแบบอัตโนมัติ เก็บผลการทดสอบตั้งแต่วันที่ 28 กันยายน พ.ศ. 2565 ถึงวันที่ 6 ตุลาคม พ.ศ. 2565 บันทึกข้อมูลทุก ๆ 1 นาที เป็นเวลา 24 ชั่วโมง ในพื้นที่ตำบลหัวรอ อำเภอเมือง จังหวัดพิษณุโลก

การเก็บข้อมูลของเซ็นเซอร์ในการวัดค่าอุณหภูมิอากาศ ความชื้นอากาศ และความเข้มแสงอาทิตย์

เก็บข้อมูลในวันที่ 28 กันยายน พ.ศ. 2565 ถึงวันที่ 2 ตุลาคม พ.ศ. 2565 บันทึกข้อมูลทุก ๆ 1 นาที เป็นเวลา 24 ชั่วโมง โดยติดตั้งอุปกรณ์เก็บตัวอย่างฝุ่นตกแบบอัจฉริยะไว้กลางแจ้งเพื่อทดสอบกับสภาพอากาศจริง ซึ่งในช่วงที่ทำการเก็บผลการทดลองเป็นช่วงที่มีพายุเข้าและเป็นช่วงที่มีโอกาสฝนตกสูง

จากภาพ 26 แสดงผลของอุณหภูมิอากาศ ความชื้นอากาศ และความเข้มแสงอาทิตย์บริเวณจุดติดตั้งอุปกรณ์วันที่ วันที่ 28 กันยายน พ.ศ. 2565 ถึงวันที่ 2 ตุลาคม พ.ศ. 2565 พบว่าในช่วงวันและเวลาดังกล่าวได้มีพายุฝนฟ้าคะนอง ส่งผลให้วันที่ 29 กันยายน พ.ศ. 2565 ถึงวันที่ 1 ตุลาคม พ.ศ. 2565 มีฝนตก ทำให้มีความชื้นอากาศสูงและอุณหภูมิอากาศลดลง โดยค่าอุณหภูมิและความชื้นอากาศสูงสุดเท่ากับ 34 ℃ และ 91 %RH ค่าอุณหภูมิและความชื้นอากาศต่ำสุดเท่ากับ 24 ℃ และ 49 %RH ตามลำดับ ค่าความเข้มแสงอาทิตย์สูงสุดเท่ากับ 518 W/m^2 ซึ่งผลการเก็บข้อมูลแสดงให้เห็นว่าอุปกรณ์ได้มีการเก็บบันทึกค่าของอุณหภูมิอากาศ ความชื้นอากาศ และความเข้มแสงอาทิตย์ได้อย่างมีประสิทธิภาพ

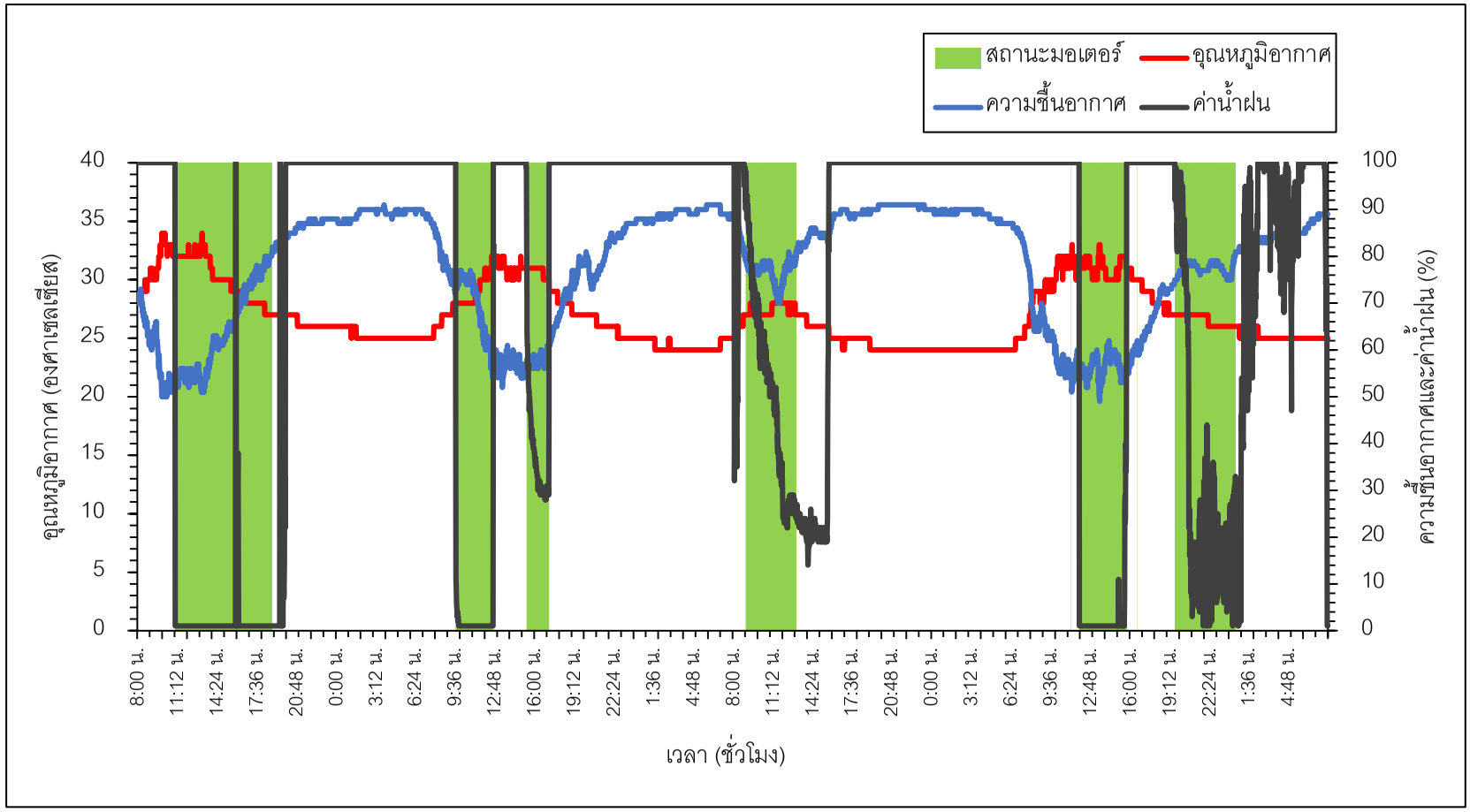


ภาพ 27 อุณหภูมิอากาศ ความชื้นอากาศ และความเข้มแสงอาทิตย์ บริเวณจุดติดตั้งอุปกรณ์วันที่ 28 กันยายน พ.ศ. 2565 ถึงวันที่ 2 ตุลาคม พ.ศ. 2565 บันทึกข้อมูลทุกๆ 1 นาที เป็นเวลา 24 ชั่วโมง

การเปิดโหมดคำสั่งการเปิดปิดฝาโหลเก็บตัวอย่างฝุ่นตกแบบอัตโนมัติ

การเปิดปิดฝาโหลเก็บตัวอย่างฝุ่นตกนั้นช่วยป้องกันน้ำฝนที่ตกลงสู่โหลเก็บตัวอย่างฝุ่นตก เพื่อไม่ให้มีน้ำในโหลเก็บตัวอย่างฝุ่นตกมากเกินไป ซึ่งส่งผลต่อการนำตัวอย่างไปวิเคราะห์และส่งผล เกิดความต่อเนื่องในการเก็บข้อมูลฝุ่นตก การทดสอบได้ทำการทำการเปิดชุดคำสั่งการเปิดปิดแบบอัตโนมัติในแอปพลิเคชัน Blynk เป็นเวลา 4 วัน ตั้งแต่วันที่ 3 ตุลาคม พ.ศ. 2565 ถึงวันที่ 6 ตุลาคม พ.ศ. 2565 บันทึกข้อมูลทุก ๆ 1 นาที เป็นเวลา 24 ชั่วโมง

ภาพ 28 แสดงสถานะของการเปิดปิดฝาโหลเก็บตัวอย่างฝุ่นตก อุณหภูมิอากาศ ความชื้นอากาศ และเซ็นเซอร์น้ำฝน บริเวณจุดติดตั้งอุปกรณ์วันที่ วันที่ 3 ตุลาคม พ.ศ. 2565 ถึงวันที่ 6 ตุลาคม พ.ศ. 2565 พบว่าในช่วงเวลาตลอดทั้ง 4 วัน ได้มีฝนตกเป็นช่วงๆ ของการบันทึกข้อมูล ซึ่งได้กำหนดค่าสถานะของการเปิดปิดฝาโหลเก็บตัวอย่างฝุ่นตกไว้ หากฝนตกฝาปิดให้มอเตอร์สถานะเป็น 0 ถ้าหากฝนไม่ตกฝาเปิดให้สถานะเป็น 1 อีกทั้งยังกำหนดค่าการเปลี่ยนแปลงของอุณหภูมิอากาศ ความชื้นอากาศ และเซ็นเซอร์น้ำฝนในแอปพลิเคชัน Blynk เพื่อให้การเปิดปิดฝาโหลเก็บตัวอย่างฝุ่นตกแบบอัตโนมัติทำได้อย่างแม่นยำมากยิ่งขึ้น โดยกำหนดให้อุณหภูมิอากาศที่อยู่ในช่วงค่าสูงกว่า 21 ℃ สั่งให้ฝาเปิด หากค่าต่ำกว่า 27 ℃ สั่งให้ฝาปิด ความชื้นอากาศที่อยู่ในช่วงค่าสูงกว่า 80 %RH สั่งให้ฝาปิด หากค่าต่ำกว่า 70 %RH สั่งให้ฝาเปิด และค่าเซ็นเซอร์น้ำฝนที่อยู่ในช่วงค่าสูงกว่า 85 % สั่งให้ฝาปิด หากค่าต่ำกว่า 90 %RH สั่งให้ฝาเปิด ซึงจากผลการทดลองแสดงให้เห็นว่ามอเตอร์ได้ทำการเปิดปิดตามค่าที่กำหนดตามค่าสถานะที่แสดงในรูปที่กล่าวมาข้างต้น

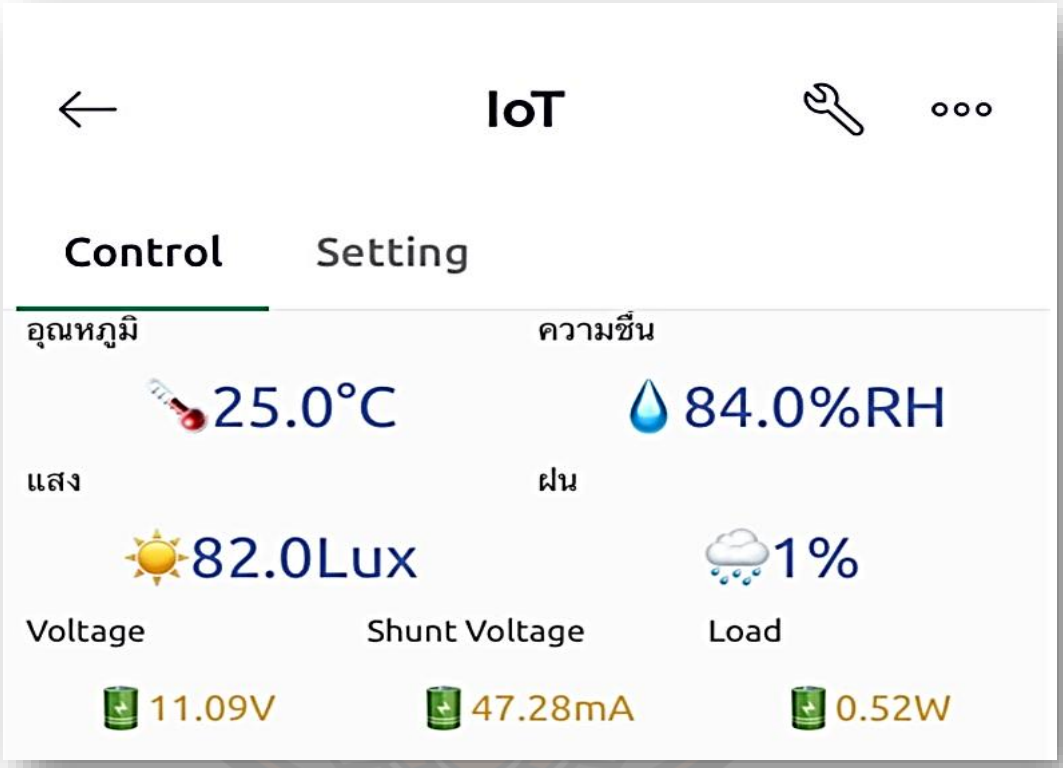


ภาพ 28 สถานะของการเปิดปิดฝาโหลเก็บตัวอย่างฝุ่นตก อุณหภูมิอากาศ ความชื้นอากาศ และเซ็นเซอร์น้ำฝน บริเวณจุดติดตั้งอุปกรณ์

วันที่ 3 ตุลาคม พ.ศ. 2565 ถึงวันที่ 6 ตุลาคม พ.ศ. 2565

การแสดงผลข้อมูลบนแอปพลิเคชัน Blynk แบบเรียลไทม์

จากภาพ 29 แสดงค่าที่ได้จากเซ็นเซอร์วัดค่าอุณหภูมิอากาศ ความชื้นอากาศ ความเข้มแสงอาทิตย์ และโมดูลวัดค่ากระแสไฟฟ้า โดยค่าต่างๆ จะแสดงบนหน้าแอปพลิเคชันแบบเรียลไทม์ เพื่อให้ได้ทราบถึงสถานะของเซ็นเซอร์และลักษณะของสถาพอากาศบริเวณพื้นที่ทำการทดสอบ ซึ่งการแสดงค่าที่ได้จากเซ็นเซอร์ในตลอดระยะเวลาการเก็บผลการทดสอบสามารถแสดงค่าข้อมูลได้อย่างต่อเนื่อง



ภาพ 29 การแสดงค่าของเซ็นเซอร์แบบเรียลไทม์บนแอปพลิเคชัน Blynk

บทที่ 5

บทสรุป

สรุปและอภิปรายผล

งานวิจัยนี้เป็นการประยุกต์ใช้เทคโนโลยีอินเทอร์เน็ตของสรรพสิ่งร่วมกับอุปกรณ์เก็บตัวอย่างฝุ่นตกแบบอัจฉริยะ ทำการติดตั้งเซ็นเซอร์วัดค่าพารามิเตอร์ และการสั่งการเปิดปิดฝาโหลเก็บตัวอย่างฝุ่นตกแบบอัตโนมัติ ในวันที่ 28 กันยายน พ.ศ. 2565 ถึงวันที่ 6 ตุลาคม พ.ศ. 2565 บันทึกข้อมูลทุก ๆ 1 นาที เป็นเวลา 24 ชั่วโมง ในพื้นที่ตำบลหัวรอ อำเภอเมือง จังหวัดพิษณุโลก จากการทดลองพบว่าการเก็บผลอุณหภูมิอากาศ ความชื้นอากาศ และความเข้มแสงอาทิตย์ ในช่วงวันที่ 29 กันยายน พ.ศ. 2565 ถึงวันที่ 1 ตุลาคม พ.ศ. 2565 เป็นช่วงที่มีพายุและฝนตกเป็นช่วง ๆ ทำให้เกิดความชื้นอากาศสูงและอุณหภูมิอากาศลดลง โดยค่าอุณหภูมิและความชื้นอากาศสูงสุดเท่ากับ 34 ℃ และ 91 %RH ค่าอุณหภูมิและความชื้นอากาศต่ำสุดเท่ากับ 24 ℃ และ 49 %RH ตามลำดับ ค่าความเข้มแสงอาทิตย์สูงสุดเท่ากับ 518 W/m^2 ซึ่งจากผลการเก็บข้อมูลแสดงให้เห็นว่าอุปกรณ์ได้มีการเก็บบันทึกค่าของอุณหภูมิอากาศ ความชื้นอากาศ และความเข้มแสงอาทิตย์ทำได้อย่างมีประสิทธิภาพ

และพบว่าการเปิดโหมดเปิดปิดฝาโหลเก็บตัวอย่างฝุ่นตกแบบอัตโนมัตินั้นช่วยป้องกันน้ำฝนที่ตกลงสู่โหลเก็บตัวอย่างฝุ่นตกช่วยไม่ให้มีน้ำในโหลเก็บตัวอย่างฝุ่นตกมากเกินไป โดยกำหนดค่าการเปลี่ยนแปลงของอุณหภูมิอากาศ ความชื้นอากาศ และเซ็นเซอร์น้ำฝนในแอปพลิเคชัน Blynk เพื่อให้การเปิดปิดฝาโหลเก็บตัวอย่างฝุ่นตกแบบอัตโนมัติทำได้อย่างแม่นยำมากยิ่งขึ้น ซึ่งการทดลองนี้ได้กำหนดอุณหภูมิอากาศที่อยู่ในช่วงค่าสูงกว่า 21 ℃ สั่งให้ฝาเปิด หากค่าต่ำกว่า 27 ℃ สั่งให้ฝาปิด ความชื้นอากาศที่อยู่ในช่วงค่าสูงกว่า 80 %RH สั่งให้ฝาปิด หากค่าต่ำกว่า 70 %RH สั่งให้ฝาเปิด และค่าเซ็นเซอร์น้ำฝนที่อยู่ในช่วงค่าสูงกว่า 85 % สั่งให้ฝาปิด หากค่าต่ำกว่า 90 %RH สั่งให้ฝาเปิด ซึ่งสถานะของการเปิดปิดฝาโหลเก็บตัวอย่างฝุ่นตกต่อการเปิดปิดฝาที่ตัวอุปกรณ์ทำได้อย่างแม่นยำ นอกจากนี้ยังพบว่าการแสดงข้อมูลบนแอปพลิเคชัน Blynk สามารถแสดงข้อมูลได้แบบเรียลไทม์ เพื่อให้ทราบถึงลักษณะของสภาพอากาศ ณ ช่วงเวลาที่ทำการทดสอบ

ข้อเสนอแนะ

งานวิจัยนี้เป็นการประยุกต์ใช้เทคโนโลยีอินเทอร์เน็ตของสรรพสิ่งร่วมกับอุปกรณ์เก็บตัวอย่างฝุ่นตกแบบอัจฉริยะ เพื่อให้เกิดการพัฒนาในด้านการนำนวัตกรรมเข้ามาร่วมใช้งานและต่อยอดในงานด้านต่าง ๆ ของการเก็บผลการทดลองทางวิทยาศาสตร์ วิศวกรรมศาสตร์ หรือด้านการแพทย์ เป็นต้น จึงมีข้อเสนอแนะดังนี้

1. การเปลี่ยนวัสดุในการทำโครงสร้างอุปกรณ์เพื่อให้มีน้ำหนักเบามากยิ่งขึ้น
2. การต่อยอดในเรื่องของธุรกิจและการจำหน่ายอุปกรณ์
3. การพัฒนาระบบเซ็นเซอร์และระบบควบคุมไฟฟ้าให้มีประสิทธิภาพมากยิ่งขึ้น
4. ควรมีการนำฝุ่นตกที่เก็บได้นำไปวิเคราะห์เทียบกับเครื่องเก็บฝุ่นตกแบบอื่น
5. ควรมีระบบแจ้งเตือนสถานะการทำงานของเครื่องและระบุตำแหน่งเครื่องได้เพื่อป้องการการสูญหาย

บรรณานุกรม

บรรณานุกรม

กรมควบคุมมลพิษ. (2547). *คู่มือการติดตามตรวจสอบคุณภาพสิ่งแวดล้อมอย่างง่าย.* กรุงเทพฯ: กรมควบคุมมลพิษ. กระทรวงทรัพยากรธรรมชาติและสิ่งแวดล้อม.

ขวัญฤทัย ทองบุญฤทธิ์. (2554). *ปริมาณโลหะหนักในฝุ่นที่ตกในเขตจังหวัดพิษณุโลก.* พิษณุโลก: สำนักหอสมุด มหาวิทยาลัยนเรศวร.

นริศรา ไทยนาม. (2552). *การแพร่กระจายของฝุ่นละอองขนาดเล็กจากโรงโม่หิน อำเภอนากลาง จังหวัดหนองบัวลำภู.* มหาสารคาม: มหาวิทยาลัยมหาสารคาม.

บวร ไชยษา. (2546). *การศึกษาปริมาณฝุ่นละอองจากการจราจรในเขตเทศบาลนครอุบลราชธานี. สำนักวิทยบริการและเทคโนโลยีสารสนเทศ.* อุบลราชธานี: มหาวิทยาลัยราชภัฏอุบลราชธานี.

ยุทธนา ตันวงศ์วาล. (บก.). (2560). *คู่มือมาตรฐานคุณภาพสิ่งแวดล้อม. สำนักงานสิ่งแวดล้อมภาคที่ 13 (ชลบุรี).* ชลบุรี: กระทรวงทรัพยากรธรรมชาติและสิ่งแวดล้อม.

ศลีจิตร น้ำจิตร, ปรมัตถ์ ราชพรหมมินทร์, พิเชษฐ์ สุขสุสาสน์, และสถาปัตย์ ชิดปราง. (2545). *การศึกษาปริมาณฝุ่นตกภายนอกอาคารบริเวณเขตเมืองพิษณุโลก.* พิษณุโลก: สำนักหอสมุด มหาวิทยาลัยนเรศวร.

สิทธิพันธ์ เสือชม, ศุภกิตติ์ ครุฑวงศ, และตรีณพ จิระวัฒน์. (2558). *ระบบควบคุมการจ่ายน้ำอัจฉริยะด้วยเทคโนโลยี IOT*. ชลบุรี: มหาวิทยาลัยบูรพา.

สำนักจัดการคุณภาพอากาศและเสียง. (2554). *รู้รอบทิศมลพิษทางอากาศ.* กรุงเทพฯ: กรมควบคุมมลพิษ. กระทรวงทรัพยากรธรรมชาติและสิ่งแวดล้อม.

อดิศักดิ์ ประพันธ์กุล, และอภินิติ โชติสังกาศ. (2563). การประยุกต์ใช้ระบบตรวจวัดเพื่อควบคุมฝุ่นในพื้นที่ก่อสร้างโดยใช้เทคโนโลยีอินเทอร์เน็ตในทุกสิ่ง. ใน *การประชุมทางวิชาการของมหาวิทยาลัยเกษตรศาสตร์ ครั้งที่ 58. มหาวิทยาลัยเกษตรศาสตร์.* กรุงเทพฯ: สำนักหอสมุด. มหาวิทยาลัยเกษตรศาสตร์.

Jerry Sartara. (2562). *มอเตอร์เกียร์ คืออะไร ต่างจากมอเตอร์ธรรมดาอย่างไร.* สืบค้น 17 กุมภาพันธ์ 2565, จาก https://jerrysartaramastores.com

PoundXI. (2561). *วิธีใช้งานโปรแกรม Arduino IDEb เบื้องต้น.* สืบค้น 17 กุมภาพันธ์ 2565, จาก https://poundxi.com

PSP TECH CO.,LTD. (2557). *รีเลย์ (Relay) คืออะไร.* สืบค้น 15 กุมภาพันธ์ 2565, จาก http://www.psptech.co.th/รีเลย์relayคืออะไร-15696.page

Blynk. (2021). *Introduction*. Retrieved February, 17, 2022, from https://docs.blynk.io/en/?_ga=2.17251641.1464725736.1645114947-2122142101.1645114947

Kanan, R., Elhassan, O., & Bensalem, R. (2018). An IoT-based autonomous system for workers' safety in construction sites with real-time alarming, monitoring, and positioning strategies. *Automation in Construction*, *88*, 73-86.

Microdigisoft. (2021). *ESP32 with MODBUS RTU RS485 Protocol Using Arduino IDE.* Retrieved February, 16, 2022, from https://microdigisoft.com/esp32-with-modbus-rtu-rs485-protocol-using-arduino-ide/

Santos, R., & Santos, S. (1998). *Guide to NEO-6M GPS Module with Arduino.* Retrieved February 16, 2022, from https://randomnerdtutorials.com/guide-to-neo-6m-gps-module-with-arduino/

Serpanos, D., & Wolf, M. (2018). *Internet-of-Things (IoT) Systems Architectures, Algorithms, Methodologies*. N.P.: Springer Nature.

Tripathy, B. K., & Anuradha, J. (Eds.). (2017). *Internet of things (IoT): technologies, applications, challenges and solutions*. Boca Raton: CRC press.

ภาคผนวก

ภาคผนวก ก ชุดคำสั่งในโปรแกรม Arduino IDE ที่ใช้ในงานวิจัยของการประยุกต์ใช้เทคโนโลยีอินเทอร์เน็ตของสรรพสิ่งร่วมกับอุปกรณ์เก็บตัวอย่างฝุ่นตกแบบอัจฉริยะ

```
#define BLYNK_TEMPLATE_ID "xxxxxxxxxx"
#define BLYNK_DEVICE_NAME "xxxxxxxxxx"
//#define BLYNK_TEMPLATE_ID "xxxxxxxxx"
//#define BLYNK_DEVICE_NAME "xxxxxx"
#define BLYNK_FIRMWARE_VERSION        "0.1.0"

#define BLYNK_PRINT Serial
//#define BLYNK_DEBUG

#define APP_DEBUG

// Uncomment your board, or configure a custom board in Settings.h
//#define USE_WROVER_BOARD
//#define USE_TTGO_T7
//#define USE_ESP32C3_DEV_MODULE
//#define USE_ESP32S2_DEV_KIT

#include "BlynkEdgent.h"

#include <SPI.h>
#include <Wire.h>
#include <Adafruit_GFX.h>
/*********************************Google Sheet*******************/
#include <HTTPClient.h>
#include "time.h"
const char* ntpServer = "pool.ntp.org";
const long  gmtOffset_sec = 19800;
const int   daylightOffset_sec = 0;
String GOOGLE_SCRIPT_ID = "xxxxxxxxxx";// change Gscript ID
int count = 0;
```

```
31
32 /************************Oled*************************/
33 #include <SSD1306Wire.h>
34 std::vector<uint8_t> img1;
35 std::vector<uint8_t> img2;
36 std::vector<uint8_t> img3;
37 std::vector<uint8_t> img4;
38 std::vector<uint8_t> img5;
39 std::vector<uint8_t> img6;
40 std::vector<uint8_t> img7;
41 SSD1306Wire oled1(0x3c, 21, 22);
42
43
44 #include <time.h>
45 #include <ModbusMaster.h>
46 #include <SoftwareSerial.h>  //https://github.com/PaulStoffregen/SoftwareSerial
47 #define TXD   // your software serial TXD pin
48 #define RXD  // your software serial RXD pin
49 #define slaveID 1
50
51 ModbusMaster node;
52
53 /************************TH-031*************************/
54 #include "TinyGPS++.h"
55 #include "SoftwareSerial.h"
56 #define HUMIDITY_REG 00
57 #define TEMPERATURE_REG 01
58 #define Light_REG 0x8
59 float Temperature, Humidity, Light;
60
61 /************************Rain Sensor*************************/
62 int sensorValue1 = 33;
63 int Value1 = 0;
```

```
int outPutValue1 = 0;

/************************Voltage/Current Sensor*************************/
#include <Adafruit_INA219.h>
Adafruit_INA219 ina219_1 (0x40);
float tagangan1, arus1 , daya1;

/************************Motor*************************/
WidgetLED ledBlynk1(V19);
int motor1Pin1 = 25;
int motor1Pin2 = 26;
int enable1Pin = 14;

// Setting PWM properties
const int freq = 30000;
const int pwmChannel = 0;
const int resolution = 10;
int dutyCycle = 2000;

bool buttonV3 = 0;
bool buttonV4 = 0;
bool buttonV9 = 0 ;
bool StatusMotor = 0;
/************************Setting*************************/
int SliderTemperatureON = 0;
int SliderHumidityON = 0;
int SliderLightON = 0;
int SliderRainON = 0;
int SliderTemperatureOFF = 0;
int SliderHumidityOFF = 0;
int SliderLightOFF = 0;
int SliderRainOFF = 0;

```

```
97  /************************BLYNK_CONNECTED *************************/
98  BlynkTimer timer;
99  BLYNK_CONNECTED() {
100   Blynk.syncAll();
101
102   oled1.display();
103   oled1.clear();
104   oled1.drawFastImage(0, 0, 128, 64, img6.data());
105   oled1.display();
106
107
108 }
109 /*****************************Setup*******************************/
110 void setup()
111 {
112   /************************Motor Pin*************************/
113   pinMode(motor1Pin1, OUTPUT);
114   pinMode(motor1Pin2, OUTPUT);
115   pinMode(enable1Pin, OUTPUT);
116   // configure LED PWM functionalitites
117   ledcSetup(pwmChannel, freq, resolution);
118   // attach the channel to the GPIO to be controlled
119   ledcAttachPin(enable1Pin, pwmChannel);
120   /************************Serial.begin*****************************/
121   Serial.begin(9600);
122   /************************Voltage/Current *************************/
123   ina219_1.begin();
124   /************************Oled *************************/
125   oled1.init();
126   oled1.flipScreenVertically();
127   oled1.setFont(ArialMT_Plain_10);
128
129   img1 = (std::vector<uint8_t> {
```

```
130     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
131     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
132     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
133     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
134     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
135     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
136     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
137     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
138     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
139     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
140     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
141     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
142     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
143     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
144     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
145     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
146     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
147     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
148     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
149     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
150     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
151     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
152     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
153     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
154     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
155     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
156     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
157     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
158     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
159     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
160     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
161     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
162     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
```

```
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xbf, 0xff, 0xff, 0xff, 0xff, 0xff, 0x3f, 0xfb, 0x5f, 0xfe,
0xff, 0xff, 0xff, 0xff, 0xbf, 0xbe, 0x7f, 0xff, 0xff, 0xff, 0xff, 0xff,
0x7f, 0xf2, 0xff, 0xff, 0xff, 0xbf, 0xf0, 0xff, 0xf7, 0xff, 0xff, 0xf5,
0xff, 0x1f, 0x87, 0xff, 0xef, 0xff, 0xff, 0xfd, 0xff, 0x67, 0x27, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xf3, 0x6f, 0xfe, 0xff, 0x07, 0x00, 0x00,
0x00, 0xf8, 0xe5, 0xf8, 0xff, 0xf1, 0xff, 0xff, 0xff, 0x3f, 0xf0, 0xfd,
0xf5, 0xf5, 0xff, 0xff, 0xff, 0x3f, 0x85, 0xfd, 0xf5, 0xbc, 0xdb, 0x5a,
0x6b, 0x5b, 0xd2, 0xf9, 0xff, 0x4e, 0x00, 0x21, 0x84, 0x84, 0xc5, 0xfd,
0xef, 0x5c, 0x57, 0x4a, 0x29, 0x29, 0xd0, 0xf9, 0xe7, 0x5d, 0xdf, 0xda,
0x6b, 0x49, 0xe5, 0xfd, 0xe7, 0xf1, 0xff, 0xff, 0xff, 0x9f, 0xea, 0xfc,
0xbf, 0x05, 0x00, 0x00, 0x00, 0x78, 0xf0, 0xee, 0x1f, 0xdf, 0xdd, 0xdd,
0xdd, 0x72, 0x7e, 0xfe, 0x9f, 0xff, 0xdd, 0xdd, 0xdd, 0xe6, 0x3f, 0xc7,
0xff, 0x3f, 0xff, 0xff, 0xff, 0x8f, 0x8f, 0xff, 0xff, 0xff, 0xbe, 0xfa,
0xff, 0x2f, 0xe0, 0xf7, 0xff, 0xff, 0xbf, 0xff, 0xff, 0xff, 0xff, 0xf9,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0x7f, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
```

```
196     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
197     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
198     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
199     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
200     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
201     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
202     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
203     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
204     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
205     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
206     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
207     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
208     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
209     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
210     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
211     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
212     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
213     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
214     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
215     0xff, 0xff, 0xff, 0xff
216   });
217   img2 = (std::vector<uint8_t> {
218     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
219     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
220     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
221     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
222     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
223     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
224     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
225     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
226     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
227     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
228     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
```

```
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0x0b, 0xa0, 0xff, 0xff, 0xff, 0xff, 0xff, 0x7f, 0x8a,
0xa7, 0xfc, 0xff, 0xff, 0xff, 0xff, 0xcf, 0xfe, 0xff, 0xed, 0xff, 0xff,
0xff, 0xff, 0xc3, 0xff, 0xff, 0xaf, 0xff, 0xff, 0xff, 0xff, 0xe0, 0xff,
0xff, 0xbf, 0xfe, 0xff, 0xff, 0x7f, 0xf8, 0xff, 0xff, 0xff, 0xfc, 0xff,
0xff, 0x9f, 0xff, 0xff, 0xff, 0xff, 0xf7, 0xff, 0xff, 0x8f, 0xff, 0xff,
0xff, 0xff, 0xe7, 0xff, 0xff, 0xc7, 0xff, 0xff, 0xff, 0xff, 0xcf, 0xff,
0xff, 0xe3, 0xff, 0xff, 0xff, 0xff, 0xbf, 0xff, 0xff, 0xf1, 0xff, 0xff,
0xff, 0xff, 0x7f, 0xff, 0xff, 0xf8, 0xff, 0xff, 0xff, 0xff, 0xff, 0xfe,
0x7f, 0xfc, 0xff, 0xff, 0xff, 0xff, 0xff, 0xf9, 0xff, 0xfe, 0xff, 0xff,
0xff, 0xff, 0xff, 0xfd, 0xbf, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0x9f, 0xff, 0xff, 0xff, 0x1f, 0xfe, 0xff, 0xf7, 0x9f, 0xff, 0xff, 0x7f,
0xfc, 0xe7, 0xff, 0xf7, 0x8f, 0xff, 0xff, 0x9f, 0x00, 0xe0, 0xff, 0xef,
0x8f, 0xff, 0xff, 0x07, 0xa0, 0x60, 0xfe, 0xff, 0x87, 0xff, 0xff, 0x1d,
0xa7, 0x28, 0xff, 0xdf, 0x87, 0xff, 0x7f, 0xd4, 0xff, 0x3f, 0xf1, 0xdf,
0x83, 0xff, 0xbf, 0xd4, 0xff, 0x7f, 0xe0, 0x3f, 0xc3, 0xff, 0x6f, 0xfc,
0xff, 0xff, 0xf0, 0x3f, 0xe3, 0xff, 0x77, 0xff, 0xff, 0xff, 0xfd, 0xbf,
0xf3, 0xff, 0xb3, 0xff, 0xff, 0xff, 0xe9, 0xff, 0xf9, 0xff, 0x91, 0xff,
0xff, 0xff, 0xff, 0x7f, 0xff, 0x7f, 0x91, 0xff, 0xfb, 0xff, 0xf3, 0x7f,
0xf9, 0xbf, 0xf0, 0xff, 0xf2, 0xff, 0xd3, 0x7f, 0xf9, 0x9f, 0xf0, 0x7f,
0xe2, 0xff, 0x97, 0x7f, 0xf9, 0x0f, 0xff, 0xff, 0xc2, 0xf0, 0x9f, 0x7f,
0xf9, 0x87, 0xff, 0xff, 0x29, 0xfa, 0xaf, 0x7f, 0xf9, 0xc3, 0xff, 0xff,
0x9b, 0xfe, 0x97, 0x7f, 0xf9, 0x9b, 0xff, 0xff, 0xf7, 0xfe, 0xe7, 0x7f,
```

```
0xf9, 0x0f, 0xff, 0xff, 0x39, 0xff, 0x1f, 0x7f, 0xf9, 0x0f, 0xf0, 0x7f,
0x18, 0xff, 0x1f, 0x7f, 0xf9, 0x1f, 0xf1, 0xff, 0x19, 0xfe, 0x13, 0x7f,
0xff, 0x3f, 0xf3, 0xff, 0x1f, 0xfc, 0x13, 0x7f, 0xf9, 0x7f, 0xe6, 0xff,
0x5f, 0xff, 0x0b, 0xff, 0xff, 0xff, 0x03, 0xff, 0x7f, 0xff, 0x09, 0x7f,
0xfb, 0xff, 0x23, 0xff, 0xff, 0xff, 0x85, 0xff, 0xfb, 0xff, 0x67, 0xfc,
0xff, 0xff, 0xc4, 0x3f, 0xff, 0xff, 0x2f, 0xe1, 0xff, 0x7f, 0xe2, 0x3f,
0xff, 0xff, 0x3f, 0xc3, 0xff, 0x3f, 0xf7, 0x7f, 0xf7, 0xff, 0xff, 0x15,
0xff, 0x8f, 0xff, 0xdf, 0xff, 0xff, 0xff, 0x77, 0x73, 0x61, 0xfe, 0xff,
0xef, 0xff, 0xff, 0x17, 0x63, 0x29, 0xfe, 0xef, 0xff, 0xff, 0xff, 0x3f,
0xf0, 0xe7, 0xff, 0xff, 0x9f, 0xff, 0xff, 0x7f, 0x00, 0xc0, 0xff, 0xf7,
0x3f, 0xff, 0xff, 0xff, 0x1f, 0xec, 0xff, 0xff, 0x3f, 0xfe, 0xff, 0xff,
0xff, 0xff, 0xff, 0xf3, 0x3f, 0xfc, 0xff, 0xff, 0xff, 0xff, 0xff, 0xf1,
0x7f, 0xfc, 0xff, 0xff, 0xff, 0xff, 0xff, 0xfe, 0xff, 0xfd, 0xff, 0xff,
0xff, 0xff, 0x7f, 0xff, 0xff, 0xfb, 0xff, 0xff, 0xff, 0xff, 0xbf, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xdf, 0xff, 0xff, 0xef, 0xff, 0xff,
0xff, 0xff, 0xef, 0xff, 0xff, 0x5f, 0xff, 0xff, 0xff, 0xff, 0xe7, 0xff,
0xff, 0x3f, 0xfc, 0xff, 0xff, 0xff, 0xe1, 0xff, 0xff, 0xbf, 0xfc, 0xff,
0xff, 0xff, 0xfa, 0xff, 0xff, 0xff, 0xfa, 0xff, 0xff, 0xbf, 0xff, 0xff,
0xff, 0xff, 0x4f, 0xfe, 0xff, 0xe7, 0xff, 0xff, 0xff, 0xff, 0x3f, 0xde,
0xff, 0xe8, 0xff, 0xff, 0xff, 0xff, 0x3f, 0x03, 0x40, 0xfb, 0xff, 0xff,
0xff, 0xff, 0xff, 0x83, 0x4f, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
```

```
295     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
296     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
297     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
298     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
299     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
300     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
301     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
302     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
303     0xff, 0xff, 0xff, 0xff
304   });
305   img3 = (std::vector<uint8_t> {
306     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
307     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
308     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
309     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
310     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
311     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
312     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
313     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
314     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
315     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
316     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
317     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
318     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
319     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
320     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
321     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
322     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
323     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
324     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
325     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
326     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
327     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
```

```
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xf3, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xcf, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xfb,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xd5, 0xff, 0x3f, 0xff, 0xff,
0xff, 0xff, 0xff, 0xef, 0xff, 0xef, 0xff, 0xff, 0xff, 0xff, 0xff, 0xb5,
0xff, 0xdb, 0xff, 0xff, 0xff, 0xff, 0xff, 0x6b, 0xfe, 0xf7, 0xff, 0xff,
0xff, 0xff, 0xff, 0xdd, 0xff, 0xec, 0xff, 0xff, 0xff, 0xff, 0xff, 0xcb,
0xbf, 0xdb, 0xff, 0xff, 0xff, 0xff, 0xff, 0xef, 0x7f, 0xf5, 0xff, 0xff,
0xff, 0xbb, 0xff, 0xff, 0xff, 0xef, 0xff, 0xff, 0xff, 0x67, 0xff, 0xbf,
0xfa, 0xf5, 0xff, 0xff, 0xff, 0xdf, 0xfa, 0xeb, 0xdf, 0xcd, 0xff, 0xff,
0xff, 0x6f, 0xff, 0x7e, 0xf5, 0xec, 0xff, 0xff, 0xff, 0xdf, 0xbc, 0xd5,
0xbf, 0xff, 0xff, 0xff, 0xff, 0xbf, 0xff, 0xbf, 0xea, 0xfd, 0xff, 0xff,
0xff, 0x7f, 0xef, 0xff, 0x5f, 0xff, 0xff, 0xff, 0xff, 0xff, 0x7c, 0xaf,
0xfb, 0xfd, 0xff, 0xff, 0xff, 0xff, 0xf6, 0xff, 0x7e, 0xcb, 0xfa, 0xff,
0xff, 0xff, 0xff, 0xab, 0xfd, 0xfe, 0x56, 0xff, 0xff, 0xff, 0xeb, 0xff,
0xff, 0xf5, 0xad, 0xfe, 0xff, 0xff, 0xbf, 0x6a, 0x7d, 0xff, 0xda, 0xff,
0xff, 0xff, 0xfb, 0xff, 0xff, 0xea, 0xf7, 0xff, 0xff, 0xff, 0xaf, 0x5a,
0xff, 0xff, 0xfa, 0xff, 0xff, 0xbf, 0xfb, 0xff, 0xff, 0xea, 0xfd, 0xff,
0xff, 0xef, 0xaf, 0xf5, 0xbf, 0xfd, 0xff, 0xff, 0xff, 0xd3, 0xfb, 0xff,
0x7f, 0xd7, 0xff, 0xff, 0xff, 0xbe, 0x77, 0xf5, 0xff, 0xfa, 0xff, 0xff,
0x7f, 0xd3, 0xdf, 0xff, 0xff, 0xef, 0xff, 0xff, 0xbf, 0xae, 0xf7, 0xfa,
0xbf, 0xfa, 0xff, 0xff, 0xbf, 0x76, 0xbf, 0xff, 0xff, 0xcf, 0xff, 0xff,
0xff, 0xf7, 0x6f, 0xfd, 0xff, 0x7a, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xfd, 0xff, 0xfe, 0xff, 0xff, 0xff, 0x5f, 0xff, 0xff, 0x5a, 0xfd, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xbf, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0x4a, 0xf9, 0xff, 0xff, 0xff, 0xf3, 0xff, 0xff, 0xff, 0xf6, 0xff,
0xff, 0xff, 0xef, 0xff, 0xff, 0xaa, 0xed, 0xff, 0xff, 0xff, 0xfb, 0xff,
0xff, 0xaf, 0xfb, 0xff, 0xff, 0xff, 0xf5, 0xff, 0xff, 0xfa, 0xfb, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xf5, 0xff,
0xfb, 0xfa, 0xff, 0xff, 0xff, 0xff, 0xfa, 0xff, 0xff, 0xfd, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xbf, 0xff, 0xff, 0xff, 0xff, 0xff, 0xfe, 0xff,
```

```
361     0xff, 0xfa, 0xff, 0xff, 0xff, 0xff, 0xfe, 0xf7, 0xff, 0xff, 0xff, 0xff,
362     0xff, 0xff, 0xff, 0xff, 0xf7, 0xfa, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
363     0xff, 0xfd, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0x7f, 0xe7, 0xff, 0xff,
364     0xff, 0xff, 0xff, 0xff, 0xff, 0xfd, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
365     0x6f, 0xfb, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xfe, 0xff, 0xff,
366     0xff, 0xff, 0xff, 0xff, 0xff, 0xfb, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
367     0xff, 0xfe, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xfd, 0xff, 0xff,
368     0xff, 0xff, 0xff, 0xff, 0xff, 0xfb, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
369     0xff, 0xfe, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
370     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
371     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
372     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
373     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
374     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
375     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
376     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
377     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
378     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
379     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
380     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
381     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
382     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
383     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
384     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
385     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
386     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
387     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
388     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
389     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
390     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
391     0xff, 0xff, 0xff, 0xff
392 });
393 img4 = (std::vector<uint8_t> {
```

```
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xfb, 0xfd,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xba, 0xe7, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xe7, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xd7, 0xff, 0xff, 0xff, 0xff,
0xff, 0xdf, 0xff, 0xb7, 0xff, 0x3f, 0xfe, 0xff, 0xff, 0x7b, 0xff, 0xbf,
0xfe, 0xcf, 0xf1, 0xff, 0xff, 0xf9, 0xfc, 0xff, 0xfd, 0x37, 0xf2, 0xff,
0xff, 0xf8, 0xfe, 0xff, 0xfd, 0xeb, 0xfb, 0xff, 0xff, 0xf9, 0xff, 0xff,
0xfb, 0x17, 0xfc, 0xff, 0x7f, 0xff, 0xff, 0xff, 0xe7, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xfb, 0xff, 0xff, 0xff, 0x7f, 0xff, 0xff, 0xff,
```

```
0xf7, 0xff, 0x7f, 0xe0, 0xff, 0xff, 0xff, 0xff, 0xe7, 0xff, 0x9f, 0xff,
0xff, 0xfe, 0xff, 0xff, 0xe7, 0xff, 0x6f, 0xe2, 0xff, 0xf0, 0xff, 0xff,
0xe7, 0xff, 0xaf, 0xfd, 0x7f, 0xfb, 0xff, 0xff, 0xfb, 0xff, 0x7f, 0xe2,
0x5f, 0xff, 0xff, 0xff, 0xff, 0xff, 0x7f, 0xfe, 0xff, 0xff, 0xff, 0xff,
0xfd, 0xbf, 0xfe, 0xff, 0xff, 0xff, 0xff, 0xff, 0xfd, 0x4f, 0xfd, 0xff,
0xf7, 0xff, 0xff, 0xff, 0xfb, 0xb3, 0xf2, 0xff, 0xff, 0xff, 0xff, 0xff,
0xf7, 0x5b, 0xf3, 0xff, 0xff, 0xff, 0xff, 0xff, 0xef, 0xa7, 0xfc, 0xff,
0xff, 0xff, 0xff, 0xff, 0x9f, 0x57, 0xfd, 0xff, 0xff, 0xff, 0xff, 0xff,
0xdf, 0xdf, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0x3f, 0xff, 0xff, 0xff,
0xfb, 0xff, 0xff, 0xff, 0xdf, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0x7f, 0xf0, 0xf7, 0xff, 0xff, 0xff, 0x9f, 0xff, 0x9f, 0xef,
0xf7, 0xff, 0xff, 0xff, 0xff, 0xff, 0x6f, 0xfa, 0xdf, 0xff, 0xff, 0xff,
0xdf, 0xff, 0x9f, 0xe5, 0xdf, 0xfe, 0xff, 0xff, 0x3f, 0xff, 0x5f, 0xea,
0x7f, 0xff, 0xff, 0xff, 0xdf, 0xff, 0x7f, 0xc4, 0xff, 0xff, 0xff, 0xff,
0xdf, 0xbf, 0xfe, 0xe5, 0xbf, 0xff, 0xff, 0xff, 0xef, 0x47, 0xfd, 0xff,
0xff, 0xff, 0xff, 0xff, 0x9f, 0xbb, 0xfe, 0xff, 0x3f, 0xff, 0xff, 0xff,
0xf7, 0x4b, 0xf9, 0xff, 0xff, 0xff, 0xff, 0xff, 0xfb, 0xb7, 0xfe, 0xff,
0xff, 0xfe, 0xff, 0xff, 0xff, 0x57, 0xfe, 0xff, 0xff, 0xfe, 0xff, 0xff,
0xfd, 0xff, 0xff, 0xfd, 0xff, 0xff, 0xff, 0xff, 0xf9, 0xff, 0xbf, 0xe2,
0xff, 0xfe, 0xff, 0xff, 0xff, 0xff, 0x5f, 0xfd, 0xff, 0xff, 0xff, 0xff,
0xf3, 0xff, 0xaf, 0xeb, 0xff, 0xff, 0xff, 0xff, 0xf3, 0xff, 0x2f, 0xe4,
0xff, 0xfc, 0xff, 0xff, 0xf3, 0xff, 0xff, 0xc9, 0xff, 0xff, 0xff, 0xff,
0xff, 0xbf, 0xfe, 0xcb, 0xff, 0xff, 0xff, 0xff, 0xfd, 0x47, 0xf9, 0xff,
0xff, 0xfd, 0xff, 0x4f, 0xff, 0xbd, 0xfe, 0xff, 0xff, 0xf3, 0xff, 0xff,
0xff, 0x63, 0xf1, 0xff, 0xff, 0xfb, 0xfd, 0x99, 0xff, 0xcf, 0xfe, 0xff,
0xff, 0x5f, 0xe7, 0xff, 0xff, 0x1f, 0xf8, 0xff, 0xff, 0x5f, 0xff, 0xf7,
0xff, 0x9f, 0xf9, 0xff, 0xff, 0xff, 0x5f, 0xf7, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0x5f, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
```

```
460     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
461     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
462     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
463     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
464     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
465     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
466     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
467     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
468     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
469     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
470     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
471     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
472     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
473     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
474     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
475     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
476     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
477     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
478     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
479     0xff, 0xff, 0xff, 0xff
480   });
481   img5 = (std::vector<uint8_t> {
482     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
483     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
484     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
485     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
486     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
487     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
488     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
489     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
490     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
491     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
492     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
```

```
493  0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
494  0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
495  0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
496  0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
497  0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
498  0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
499  0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
500  0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
501  0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
502  0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xfe, 0xff, 0xff,
503  0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
504  0xff, 0xfc, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xfc, 0x59, 0x55,
505  0x55, 0x55, 0xf5, 0xff, 0xff, 0xff, 0x5f, 0x55, 0x55, 0x55, 0xf5, 0xff,
506  0xff, 0xfe, 0xff, 0xff, 0xff, 0xff, 0x3f, 0xff, 0xff, 0xff, 0xff, 0xff,
507  0xff, 0xff, 0xff, 0xff, 0xff, 0xfc, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
508  0xff, 0xfc, 0xff, 0xff, 0xff, 0xff, 0x3f, 0xff, 0xff, 0xff, 0xff, 0xff,
509  0xff, 0xff, 0x7f, 0xff, 0x5f, 0xfc, 0xf9, 0xff, 0xff, 0xff, 0x3f, 0xff,
510  0x4f, 0xfc, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0x0f, 0xfc, 0xff, 0xff,
511  0xff, 0xff, 0xff, 0xff, 0x0f, 0xfc, 0xff, 0xff, 0xff, 0xff, 0x3f, 0xff,
512  0x0f, 0xfc, 0xf9, 0xff, 0xff, 0xff, 0x7f, 0xff, 0x0f, 0xfc, 0xff, 0xff,
513  0xff, 0xff, 0x3f, 0xff, 0x0f, 0xfe, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
514  0x0f, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0x8f, 0xff, 0xff, 0xff,
515  0xff, 0xff, 0x3f, 0xff, 0xcf, 0xfc, 0xf9, 0xff, 0xff, 0xff, 0x7f, 0xff,
516  0x7f, 0xfc, 0xff, 0xff, 0xff, 0xff, 0x3f, 0xff, 0x7f, 0xff, 0xff, 0xff,
517  0xff, 0xff, 0xff, 0xff, 0xff, 0xf8, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
518  0xff, 0xf8, 0xff, 0xff, 0xff, 0xff, 0x3f, 0xff, 0xff, 0xff, 0xff, 0xff,
519  0xff, 0xff, 0xff, 0xff, 0xff, 0xfc, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
520  0xff, 0xfc, 0xff, 0xff, 0x3f, 0xff, 0x3f, 0xff, 0xff, 0xff, 0xf9, 0xff,
521  0xf7, 0xff, 0xff, 0xff, 0xff, 0xfc, 0xff, 0xff, 0x7e, 0xff, 0xff, 0xff,
522  0xff, 0xfc, 0xff, 0xdf, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xcf,
523  0x7f, 0xff, 0x3f, 0xff, 0xff, 0xfc, 0xff, 0x8f, 0xff, 0x0d, 0xfe, 0xff,
524  0xff, 0xfc, 0xf9, 0x8f, 0x7f, 0x7e, 0xfe, 0xff, 0xff, 0xff, 0xff, 0x8f,
525  0x7f, 0x1e, 0xff, 0xff, 0xff, 0xfc, 0xff, 0x0f, 0xf9, 0x9f, 0x3f, 0xff,
```

```
0xff, 0xfc, 0xff, 0x0f, 0xf8, 0xf3, 0xff, 0xff, 0xff, 0xff, 0xff, 0x0f,
0xf8, 0xf1, 0xff, 0xff, 0xff, 0xfc, 0xff, 0x7f, 0x7f, 0xf8, 0xff, 0xff,
0xff, 0xfc, 0xff, 0xff, 0x3f, 0xfe, 0x3f, 0xff, 0xff, 0xff, 0xff, 0xff,
0x0f, 0xff, 0xff, 0xff, 0xff, 0xfc, 0xff, 0xff, 0xcf, 0xff, 0xff, 0xff,
0xff, 0xfc, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0x3f, 0xff, 0xff, 0xfc, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xfc, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0x1f, 0xfe, 0xff, 0xff, 0xff, 0xff, 0x3f, 0xff,
0x0f, 0xfe, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0x0f, 0xfe, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0x0f, 0xfe, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0x0f, 0xfe, 0xff, 0xff, 0xff, 0xff, 0x3f, 0xff, 0x0f, 0xfe, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0x0f, 0xfe, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0x0f, 0xfe, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0x0f, 0xfe, 0xff, 0xff,
0xff, 0xff, 0x3f, 0xff, 0x0f, 0xfc, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xcf, 0xfc, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xfc, 0xff, 0xff, 0xff, 0xff, 0x3f, 0xff,
0xff, 0xfc, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xfe, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xdf, 0xff, 0xff, 0xfe, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xfe, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0x01, 0xfe, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0x03, 0xfe, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
```

```
559     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
560     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
561     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
562     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
563     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
564     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
565     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
566     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
567     0xff, 0xff, 0xff, 0xff
568   });
569   img6 = (std::vector<uint8_t> {
570     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
571     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
572     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
573     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
574     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
575     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
576     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
577     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0x07, 0xfc, 0xff, 0xff,
578     0xff, 0xff, 0xff, 0xff, 0x07, 0xec, 0xff, 0xff, 0xff, 0xff, 0xff, 0x7f,
579     0x07, 0xa8, 0xff, 0xff, 0xff, 0xff, 0xff, 0x1f, 0x19, 0x26, 0xff, 0xff,
580     0xff, 0xff, 0xff, 0x0f, 0xf8, 0x9f, 0xf9, 0xff, 0xff, 0xff, 0xff, 0x87,
581     0xf8, 0xdf, 0xfd, 0xff, 0xff, 0xff, 0xff, 0x83, 0xff, 0xff, 0xff, 0xff,
582     0xff, 0xff, 0xff, 0x83, 0xff, 0xff, 0xf3, 0xff, 0xff, 0xff, 0xff, 0xf1,
583     0xff, 0xff, 0xc1, 0xff, 0xff, 0xff, 0xff, 0xff, 0x1f, 0xfe, 0x81, 0xff,
584     0xff, 0xff, 0xff, 0xfa, 0x07, 0xc0, 0x87, 0xff, 0xff, 0xff, 0xff, 0xff,
585     0xe1, 0x81, 0x87, 0xff, 0xff, 0xff, 0x7f, 0xfc, 0xe9, 0x05, 0xc7, 0xff,
586     0xff, 0xff, 0x7f, 0xfe, 0xfc, 0x17, 0x8f, 0xff, 0xff, 0xff, 0xff, 0x3e,
587     0xfe, 0x9f, 0xbe, 0xff, 0xff, 0xff, 0x3f, 0x7e, 0xfc, 0x3f, 0xbe, 0xff,
588     0xff, 0xff, 0x3f, 0x3f, 0xfe, 0x3f, 0x1e, 0xfe, 0xff, 0xff, 0x3f, 0x7c,
589     0xff, 0x7f, 0x9f, 0xff, 0xff, 0xff, 0xbf, 0xff, 0xff, 0x7f, 0x1c, 0xfe,
590     0xff, 0xff, 0xff, 0xff, 0xff, 0x7f, 0x3f, 0xfe, 0xff, 0xff, 0xff, 0xff,
591     0xff, 0xff, 0x1f, 0xfe, 0xff, 0xff, 0xff, 0x3f, 0xfe, 0xff, 0x7f, 0xfe,
```

```
592  0xff, 0xff, 0xff, 0xbf, 0xff, 0xff, 0x7f, 0xfe, 0xff, 0xff, 0xff, 0x3f,
593  0xff, 0x3f, 0x1e, 0xfe, 0xff, 0xff, 0x3f, 0x7c, 0xff, 0x1f, 0x3e, 0xfe,
594  0xff, 0xff, 0x1f, 0x7f, 0xfe, 0x0f, 0x1e, 0xfe, 0xff, 0xff, 0x0f, 0x7c,
595  0xfe, 0x0f, 0x3f, 0xfe, 0xff, 0xff, 0x87, 0xfe, 0xfc, 0xdf, 0x1f, 0xfe,
596  0xff, 0xff, 0x83, 0xff, 0xf9, 0xc7, 0x7f, 0xfe, 0xff, 0xff, 0x81, 0xff,
597  0xe1, 0xc1, 0x7f, 0xfe, 0xff, 0xff, 0x80, 0xff, 0x05, 0xc0, 0x1f, 0xfe,
598  0xff, 0x7f, 0xc0, 0xff, 0x1f, 0xe6, 0x3f, 0xfe, 0xff, 0x7f, 0xe0, 0xff,
599  0x1f, 0xfe, 0x1f, 0xfe, 0xff, 0x3f, 0xfa, 0xff, 0xff, 0xff, 0x7f, 0xfe,
600  0xff, 0x3f, 0xf8, 0xff, 0xff, 0xff, 0x7f, 0xfe, 0xff, 0x1f, 0x7c, 0x00,
601  0x00, 0x00, 0x1e, 0xfe, 0xff, 0x9f, 0x7f, 0x00, 0x00, 0x80, 0x3f, 0xfe,
602  0xff, 0x8f, 0x7f, 0x20, 0x42, 0x02, 0x1f, 0xfe, 0xff, 0xcf, 0x7f, 0x04,
603  0x08, 0x90, 0x7f, 0xfe, 0xff, 0xcf, 0x7f, 0x90, 0x00, 0x00, 0x7f, 0xfe,
604  0xff, 0xc7, 0x7f, 0x00, 0x44, 0x84, 0x1f, 0xfe, 0xff, 0xe7, 0x7f, 0x00,
605  0x00, 0x00, 0x3f, 0xfe, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0x1f, 0xfe,
606  0xff, 0xef, 0xff, 0xff, 0xff, 0xff, 0x7f, 0xfe, 0xff, 0xe3, 0xff, 0xff,
607  0xff, 0xff, 0x7f, 0xfe, 0xff, 0xf3, 0xff, 0xff, 0xff, 0xff, 0x1f, 0xfe,
608  0xff, 0xc3, 0xff, 0x3f, 0x00, 0xe0, 0x3f, 0xfe, 0xff, 0xf3, 0xff, 0x1f,
609  0x00, 0xc0, 0x1f, 0xfe, 0xff, 0xf3, 0xff, 0x0f, 0x92, 0x80, 0x7f, 0xfe,
610  0xff, 0xf3, 0xff, 0x4f, 0x00, 0xa4, 0x7f, 0xfe, 0xff, 0xf3, 0xff, 0x07,
611  0x00, 0x00, 0x1e, 0xfe, 0xff, 0xf3, 0xff, 0x8f, 0x6d, 0x83, 0x3e, 0xfe,
612  0xff, 0xf3, 0xff, 0x07, 0x69, 0x0b, 0x1e, 0xfe, 0xff, 0xe3, 0xff, 0x4f,
613  0xb5, 0x86, 0x3f, 0xfe, 0xff, 0xf3, 0xff, 0x07, 0x00, 0x20, 0x1e, 0xfc,
614  0xff, 0xef, 0xff, 0x8f, 0x00, 0x80, 0x3f, 0xfe, 0xff, 0xef, 0xff, 0x0f,
615  0x90, 0x88, 0x1f, 0xfe, 0xff, 0xff, 0xff, 0x1f, 0x02, 0xc0, 0x7f, 0xfe,
616  0xff, 0x87, 0xff, 0x3f, 0x00, 0xe0, 0x7f, 0xfe, 0xff, 0x87, 0xff, 0x3f,
617  0xfe, 0xef, 0x1f, 0xfe, 0xff, 0x5f, 0x7f, 0x00, 0xff, 0xff, 0x3f, 0xfe,
618  0xff, 0x0f, 0x7f, 0xe4, 0xff, 0xff, 0x1f, 0xfe, 0xff, 0x0f, 0x7f, 0xf0,
619  0xff, 0xff, 0x7f, 0xfe, 0xff, 0xbf, 0x7f, 0xe0, 0xff, 0xff, 0x7f, 0xfe,
620  0xff, 0x1f, 0x7f, 0xc4, 0xff, 0xff, 0x1f, 0xfe, 0xff, 0x7f, 0x7c, 0x00,
621  0x00, 0x80, 0x3f, 0xfe, 0xff, 0x3f, 0x7e, 0x10, 0x44, 0x00, 0x1f, 0xfe,
622  0xff, 0x7f, 0x7e, 0x84, 0x00, 0x92, 0x7f, 0xfe, 0xff, 0x3f, 0x7e, 0x00,
623  0x02, 0x00, 0x7f, 0xfe, 0xff, 0x3f, 0x7c, 0x20, 0x90, 0x88, 0x1f, 0xfe,
624  0xff, 0x3f, 0x7f, 0x04, 0x01, 0x00, 0x3e, 0xfe, 0xff, 0x7f, 0x7f, 0x00,
```

```
625     0x00, 0x00, 0x1e, 0xfe, 0xff, 0x1f, 0x7e, 0xd2, 0xb6, 0xed, 0x7f, 0xfe,
626     0xff, 0x1f, 0x7e, 0xc0, 0xb6, 0xed, 0x7f, 0xfe, 0xff, 0x9f, 0x7f, 0xc0,
627     0xff, 0xff, 0x1f, 0xfe, 0xff, 0x9f, 0x7f, 0xf0, 0xff, 0xff, 0x3f, 0xfe,
628     0xff, 0x1f, 0x7e, 0xe0, 0xff, 0xff, 0x1f, 0xfe, 0xff, 0x9f, 0xff, 0xff,
629     0xff, 0xff, 0x7f, 0xfe, 0xff, 0x9f, 0xff, 0xff, 0xff, 0xff, 0x7f, 0xfe,
630     0xff, 0x9f, 0xff, 0xff, 0x7f, 0xf8, 0x1f, 0xfe, 0xff, 0x1f, 0xff, 0xff,
631     0x5f, 0xe3, 0x7f, 0xfe, 0xff, 0x9f, 0xfe, 0xff, 0x1f, 0xeb, 0x1f, 0xfe,
632     0xff, 0x1f, 0xfc, 0xff, 0x1f, 0x9e, 0x9f, 0xff, 0xff, 0x1f, 0xfc, 0xff,
633     0x07, 0x58, 0x1f, 0xfe, 0xff, 0x7f, 0xfe, 0xff, 0x4f, 0x30, 0xdf, 0xff,
634     0xff, 0x7f, 0xff, 0xff, 0x0f, 0xb0, 0xbf, 0xff, 0xff, 0x3f, 0xff, 0xff,
635     0x37, 0xb2, 0x0f, 0xff, 0xff, 0x3f, 0xfe, 0xbf, 0x48, 0xf0, 0x0f, 0xff,
636     0xff, 0xff, 0xfe, 0x1f, 0x69, 0x98, 0x4f, 0xff, 0xff, 0xff, 0xf8, 0x6f,
637     0xd3, 0xdc, 0xc7, 0xff, 0xff, 0x7f, 0xf8, 0xef, 0x37, 0xec, 0xe7, 0xff,
638     0xff, 0xff, 0xfd, 0x7f, 0x86, 0xe7, 0xe3, 0xff, 0xff, 0xff, 0xf8, 0x27,
639     0x7c, 0xe8, 0xc3, 0xff, 0xff, 0xff, 0xe1, 0x4f, 0xf6, 0xfc, 0xe1, 0xff,
640     0xff, 0xff, 0xf1, 0x6f, 0xc7, 0xfc, 0xf8, 0xff, 0xff, 0xff, 0xe1, 0x4f,
641     0xeb, 0xff, 0xf8, 0xff, 0xff, 0xff, 0xd7, 0xbf, 0xf9, 0xff, 0xfd, 0xff,
642     0xff, 0xff, 0x9f, 0x9f, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0x4f, 0xbf,
643     0xfe, 0xff, 0xff, 0xff, 0xff, 0xff, 0x6f, 0xf9, 0xfe, 0x77, 0xff, 0xff,
644     0xff, 0xff, 0x3f, 0xf9, 0xff, 0x41, 0xff, 0xff, 0xff, 0xff, 0xbf, 0xc8,
645     0x3f, 0xe0, 0xff, 0xff, 0xff, 0xff, 0xff, 0xd1, 0x00, 0xf8, 0xff, 0xff,
646     0xff, 0xff, 0xff, 0xc9, 0x01, 0xfe, 0xff, 0xff, 0xff, 0xff, 0xff, 0x7f,
647     0xc0, 0xfe, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xc0, 0xff, 0xff, 0xff,
648     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
649     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
650     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
651     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
652     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
653     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
654     0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
655     0xff, 0xff, 0xff, 0xff
656   });
657   BlynkEdgent.begin();
```

```
  timer.setInterval(30000L, get_LTH_TM);
  timer.setInterval(60000L, googlesheet);
}

/***************************Loop*****************************/
void loop() {
  BlynkEdgent.run();
  timer.run();

}
/***************************google sheet*****************************/
void googlesheet() {
  if (WiFi.status() == WL_CONNECTED) {
    static bool flag = false;
    String urlFinal = "https://script.google.com/macros/s/" + GOOGLE_SCRIPT_ID +
"/exec?"
                      + "temperature=" + String(Temperature, 1)
                      + "&humidity=" + String(Humidity, 1)
                      + "&light=" + String(Light, 1)
                      + "&rain=" + String(outPutValue1)
                      + "&modeAuto=" + String(buttonV9)
                      + "&motorControl=" + String(StatusMotor)
                      + "&batteryV=" + String(tagangan1)
                      + "&batterymA=" + String(arus1)
                      + "&batteryW=" + String(daya1);

    Serial.print("POST data to spreadsheet:");
    Serial.println(urlFinal);
    HTTPClient http;
    http.begin(urlFinal.c_str());
    http.setFollowRedirects(HTTPC_STRICT_FOLLOW_REDIRECTS);
    int httpCode = http.GET();
    Serial.print("HTTP Status Code: ");
```

```
690     Serial.println(httpCode);
691     //-------------------------------------------------------------------
692     //getting response from google sheet
693     String payload;
694     if (httpCode > 0) {
695       payload = http.getString();
696       Serial.println("Payload: " + payload);
697     }
698     //-------------------------------------------------------------------
699     http.end();
700   }
701 }
702 /*************************TH-031 **************************/
703 int get_modbusRTU_sensor(int addr , uint16_t  REG) {
704   int result;
705   node.begin(addr, Serial);
706   result = node.readInputRegisters (REG, 4);
707   result = node.readHoldingRegisters (REG, 4);
708   if (result == node.ku8MBSuccess) {
709     return node.getResponseBuffer(0);
710   } else {
      //Serial.print("Connec modbus fail. REG >>> "); mySerial.println(REG, HEX); //
711 Debug
712
713   }
714 }
715
716 /*************************TH-031 **************************/
717 void get_LTH_TM() {
718
719   Serial.begin(9600);
720   Humidity = get_modbusRTU_sensor(slaveID, HUMIDITY_REG) / 10;
721   Temperature = get_modbusRTU_sensor(slaveID, TEMPERATURE_REG) / 10;
```

```
722   Light = get_modbusRTU_sensor(slaveID, Light_REG);
723   Serial.print("\n");
724   Serial.print("Temperature = " + String(Temperature, 1) + " C" + "\n");
725   Serial.print("Humidity = " + String(Humidity, 1) + " %RH" + "\n");
726   Serial.print("Light = " + String(Light, 1) + " lux" + "\n");
727
728   Blynk.virtualWrite(V0, String(Temperature, 1));
729   Blynk.virtualWrite(V1, String(Humidity, 1));
730   Blynk.virtualWrite(V2, String(Light, 1));
731
732   Value1 = analogRead(sensorValue1);
733   outPutValue1 = map(Value1, 0, 4095, 100, 0);
734   Serial.print("Rain = " + String(outPutValue1) + "%" + "\n");
735   Blynk.virtualWrite(V8, String(outPutValue1));
736   DC();
737
738 }
739
740
741 /************************Oled*************************/
742 void Oled() {
743   oled1.display();
744   oled1.clear();
745   oled1.drawFastImage(0, 0, 128, 64, img1.data());
746   oled1.display();
747   oled1.clear();
748   delay(2000);
749   oled1.setFont(ArialMT_Plain_16);
750   oled1.drawString(30, 16, " Temperature");
751   oled1.setFont(ArialMT_Plain_16);
752   oled1.drawString(30, 32, " " + String(Temperature, 1) + " °C");
753   oled1.display();
754   oled1.clear();
```

```
755   delay(2000);
756   oled1.drawFastImage(0, 0, 128, 64, img2.data());
757   oled1.display();
758   oled1.clear();
759   delay(2000);
760   oled1.setFont(ArialMT_Plain_16);
761   oled1.drawString(30, 16, " Humidity");
762   oled1.setFont(ArialMT_Plain_16);
763   oled1.drawString(30, 32, " " + String(Humidity, 1) + " %RH");
764   oled1.display();
765   oled1.clear();
766   delay(2000);
767   oled1.drawFastImage(0, 0, 128, 64, img3.data());
768   oled1.display();
769   oled1.clear();
770   delay(2000);
771   oled1.setFont(ArialMT_Plain_16);
772   oled1.drawString(30, 16, " Light");
773   oled1.setFont(ArialMT_Plain_16);
774   oled1.drawString(30, 32, " " + String(Light, 1) + " lux");
775   oled1.display();
776   oled1.clear();
777   delay(2000);
778   oled1.drawFastImage(0, 0, 128, 64, img4.data());
779   oled1.display();
780   oled1.clear();
781   delay(2000);
782   oled1.setFont(ArialMT_Plain_16);
783   oled1.drawString(30, 16, " Rain");
784   oled1.setFont(ArialMT_Plain_16);
785   oled1.drawString(30, 32, " " + String(outPutValue1) + "%");
786   oled1.display();
787   oled1.clear();
```

```
788   delay(2000);
789   oled1.drawFastImage(0, 0, 128, 64, img5.data());
790   oled1.display();
791   oled1.clear();
792   delay(2000);
793   oled1.setFont(ArialMT_Plain_16);
794   oled1.drawString(30, 0, "Battery");
795   oled1.setFont(ArialMT_Plain_10);
796   oled1.drawString(20, 15, "Voltage:" + String(tagangan1) + " V");
797   oled1.setFont(ArialMT_Plain_10);
798   oled1.drawString(20, 27, "Shunt Voltage:" + String(arus1) + " mA");
799   oled1.setFont(ArialMT_Plain_10);
800   oled1.drawString(20, 37, "Load Voltage:" + String(daya1) + " W");
801   oled1.display();
802   oled1.clear();
803   delay(2000);
804   oled1.drawFastImage(0, 0, 128, 64, img6.data());
805   oled1.display();
806   oled1.clear();
807   delay(2000);
808
809 }
    /*****************************Voltage/Current
810 Sensor**********************************/
811 void DC() {
812   tagangan1 = ina219_1.getBusVoltage_V();
813   arus1 = ina219_1.getShuntVoltage_mV();
814   daya1 = tagangan1 * (arus1 / 1000);
815
816   Serial.print("Voltage:    "); Serial.print(tagangan1); Serial.println(" V");
817   Serial.print("Shunt Voltage: "); Serial.print(arus1); Serial.println(" mV");
818   Serial.print("Load Voltage:  "); Serial.print(daya1); Serial.println(" Watt");
819   Serial.println("---------------------------------");
```

```
820
821   Blynk.virtualWrite(V12, String(tagangan1));
822   Blynk.virtualWrite(V13, String(arus1));
823   Blynk.virtualWrite(V20, String(daya1));
824
825 }
826 /*****************************Mode Auto*******************************/
827 void Auto() {
828
829   Serial.println("AUTOON1");
830
831   if (StatusMotor == 0) {
832     Serial.println(String(StatusMotor));
833     if (Temperature >= SliderTemperatureON
834         && Humidity <= SliderHumidityON
835         /*&& Light >= SliderLightON*/
836         && outPutValue1 <= SliderRainON) {
837       Serial.println("ON");
838       ledBlynk1.on();
839       Serial.println("MotorON");
840       digitalWrite(motor1Pin1, LOW);
841       digitalWrite(motor1Pin2, HIGH);
842       delay(1500);
843       digitalWrite(motor1Pin1, LOW);
844       digitalWrite(motor1Pin2, LOW);
845       StatusMotor = 1;
846       // Move DC motor forward with increasing speed
847
848       while (dutyCycle <= 2000) {
849         ledcWrite(pwmChannel, dutyCycle);
850         Serial.print("Forward with duty cycle: ");
851         Serial.println(dutyCycle);
852         dutyCycle = dutyCycle + 5;
```

```
853         delay(500);
854       }
855       dutyCycle = 2000;
856     }
857   }
858
859
860   if (StatusMotor == 1) {
861     Serial.println(String(StatusMotor));
862     if (Temperature <= SliderTemperatureOFF
863         && Humidity >= SliderHumidityOFF
864         /*&& Light <= SliderLightOFF*/
865         && outPutValue1 >= SliderRainOFF) {
866       Serial.println("OFF");
867       ledBlynk1.off();
868       Serial.println("MotorOFF");
869       digitalWrite(motor1Pin1, HIGH);
870       digitalWrite(motor1Pin2, LOW);
871       delay(1500);
872       digitalWrite(motor1Pin1, LOW);
873       digitalWrite(motor1Pin2, LOW);
874       StatusMotor = 0 ;
875       // Move DC motor forward with increasing speed
876
877       while (dutyCycle <= 2000) {
878         ledcWrite(pwmChannel, dutyCycle);
879         Serial.print("Forward with duty cycle: ");
880         Serial.println(dutyCycle);
881         dutyCycle = dutyCycle + 5;
882         delay(500);
883       }
884       dutyCycle = 2000;
885
```

```
886       }
887     }
888   }
889   /************************Motor Open*************************/
890   BLYNK_WRITE(V3) {
891
892     buttonV3 = param.asInt();
893
894     if (buttonV3 == 1) {
895       if (StatusMotor == 0) {
896         ledBlynk1.on();
897         Serial.println("MotorON");
898         digitalWrite(motor1Pin1, LOW);
899         digitalWrite(motor1Pin2, HIGH);
900         delay(1500);
901         digitalWrite(motor1Pin1, LOW);
902         digitalWrite(motor1Pin2, LOW);
903         // Move DC motor forward with increasing speed
904
905         while (dutyCycle <= 2000) {
906           ledcWrite(pwmChannel, dutyCycle);
907           Serial.print("Forward with duty cycle: ");
908           Serial.println(dutyCycle);
909           dutyCycle = dutyCycle + 5;
910           delay(500);
911         }
912         dutyCycle = 2000;
913         StatusMotor = 1;
914       }
915     }
916     if (buttonV3 == 0) {
917       Serial.println("Moving Backwards");
918       digitalWrite(motor1Pin1, LOW);
```

```
    digitalWrite(motor1Pin2, LOW);
    delay(500);
    // Move DC motor forward with increasing speed

    while (dutyCycle <= 2000) {
      ledcWrite(pwmChannel, dutyCycle);
      Serial.print("Forward with duty cycle: ");
      Serial.println(dutyCycle);
      dutyCycle = dutyCycle + 5;
      delay(500);
    }
    dutyCycle = 2000;
  }

}

/************************Motor Close**************************/
BLYNK_WRITE(V4) {
  buttonV4 = param.asInt();
  if (buttonV4 == 1) {
    if (StatusMotor == 1) {
      ledBlynk1.off();
      Serial.println("MotorOFF");
      digitalWrite(motor1Pin1, HIGH);
      digitalWrite(motor1Pin2, LOW);
      delay(1500);
      digitalWrite(motor1Pin1, LOW);
      digitalWrite(motor1Pin2, LOW);
      // Move DC motor forward with increasing speed

      while (dutyCycle <= 2000) {
        ledcWrite(pwmChannel, dutyCycle);
```

```
952         Serial.print("Forward with duty cycle: ");
953         Serial.println(dutyCycle);
954         dutyCycle = dutyCycle + 5;
955         delay(500);
956       }
957       dutyCycle = 2000;
958       StatusMotor = 0 ;
959     }
960   }
961   if (buttonV4 == 0) {
962     Serial.println("Moving Backwards");
963     digitalWrite(motor1Pin1, LOW);
964     digitalWrite(motor1Pin2, LOW);
965     delay(500);
966     // Move DC motor forward with increasing speed
967
968     while (dutyCycle <= 2000) {
969       ledcWrite(pwmChannel, dutyCycle);
970       Serial.print("Forward with duty cycle: ");
971       Serial.println(dutyCycle);
972       dutyCycle = dutyCycle + 5;
973       delay(500);
974     }
975
976   }
977 }
978
979 /*************************Set Temperature ON*************************/
980 BLYNK_WRITE(V5) {
981   SliderTemperatureON = param.asInt();
982 }
983
984 /************************Set Humidity ON*************************/
```

```
985 BLYNK_WRITE(V6) {
986   SliderHumidityON = param.asInt();
987 }
988
989 /************************Set Light ON*************************/
990 BLYNK_WRITE(V7) {
991   SliderLightON = param.asInt();
992
993 }
994
995 /************************Set Rain ON*************************/
996 BLYNK_WRITE(V14) {
997   SliderRainON = param.asInt();
998 }
999
1000 /************************Set Temperature OFF*************************/
1001 BLYNK_WRITE(V17) {
1002   SliderTemperatureOFF = param.asInt();
1003 }
1004
1005 /************************Set Humidity OFF*************************/
1006 BLYNK_WRITE(V15) {
1007   SliderHumidityOFF = param.asInt();
1008 }
1009
1010 /************************Set Light OFF*************************/
1011 BLYNK_WRITE(V16) {
1012   SliderLightOFF = param.asInt();
1013 }
1014
1015 /************************Set Rain OFF*************************/
1016 BLYNK_WRITE(V18) {
1017   SliderRainOFF = param.asInt();
```

```
1018 }
1019
1020 /************************Mode Auto************************/
1021 BLYNK_WRITE(V9) {
1022   buttonV9 = param.asInt();
1023   if (buttonV9 == 1) {
1024     Auto();
1025     Blynk.syncVirtual(V9);
1026   }
1027   if (buttonV9 == 0) {
1028   }
1029 }
1030 /************************ Open OLED************************/
1031 BLYNK_WRITE(V10) {
1032   if (param.asInt() == 1) {
1033     Serial.println("Oled");
1034     Oled();
1035     Blynk.syncVirtual(V10);
1036   }
1037   if (param.asInt() == 0) {
1038   }
1039 }
****** //Arrangement  by Chirawat Thongkaemkaeo. ********
```

ภาคผนวก ข การตีพิมพ์ผลงานวิทยานิพนธ์ การประชุมวิชาการระดับชาติ ครั้งที่ 19
มหาวิทยาลัยเกษตรศาสตร์ วิทยาเขตกำแพงแสน

การประยุกต์ใช้เทคโนโลยีอินเทอร์เน็ตของสรรพสิ่งเพื่อออกแบบและพัฒนาอุปกรณ์เก็บตัวอย่างฝุ่นตก (Dust-fall Jar) แบบอัจฉริยะ

Intelligent Dust-fall jar with Internet of Things Application

จีรวัฒน์ ทองแกมแก้ว[1] ปาจรีย์ ทองสนิท[1] สุภาวรรณ ศรีรัตนา[1] และสมชาย เจียจิตต์สวัสดิ์[2]

Chirawat Thongkaemkaeo [1], Pajaree Thongsanit [1], Supawan Srirattana [1] , and Somchai Jiajitsawat[2]

บทคัดย่อ

งานวิจัยนี้มีวัตถุประสงค์เพื่อออกแบบและพัฒนาอุปกรณ์เก็บตัวอย่างฝุ่นตก (Dust-fall Jar) แบบอัจฉริยะ โดยมีวัตถุประสงค์หลักเพื่อรองรับการใช้งานในช่วงฤดูฝนที่มักมีน้ำฝนตกลงสู่ชุดอุปกรณ์เก็บตัวอย่างเป็นจำนวนมาก สร้างความยากลำบากให้กับผู้ทำการเก็บตัวอย่าง โดยระบบอัจฉริยะที่ถูกติดตั้งเข้าไปเพื่อเพิ่มศักยภาพการทำงานของอุปกรณ์ ได้แก่ ไมโครคอนโทรลเลอร์, เซ็นเซอร์น้ำฝน, เซ็นเซอร์วัดความเข้มแสง-อุณหภูมิ-และความชื้น แบบกลางแจ้ง, ชุดขับมอเตอร์, มอเตอร์เกียร์ และระบบโซลาเซลล์แบบออฟกริด ผลการทดสอบพบว่าระบบสามารถทำงานได้อย่างมีประสิทธิภาพ ต่อเนื่อง แม่นยำ สอดคล้องกับสภาวะความเป็นจริงที่เกิดขึ้น โดยเมื่อเกิดสภาวะฝนตก เซ็นเซอร์น้ำฝนจะทำหน้าที่ตรวจจับค่าและสั่งการระบบให้ทำการปิดฝาของโหลเก็บตัวอย่างแบบอัตโนมัติ และสั่งให้ระบบเปิดฝาอีกครั้งเมื่อฝนหยุดลง นอกจากนี้ผู้ควบคุมยังสามารถตรวจค่าสถานะทุกอย่างของระบบผ่านแอปพลิเคชัน Blynk ได้ทุกที่และทุกเวลาอีกด้วย

คำสำคัญ : อินเทอร์เน็ตของสรรพสิ่ง, อุปกรณ์เก็บตัวอย่างฝุ่นตก, อัจฉริยะ

[1] ภาควิชาวิศวกรรมโยธา คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร จ.พิษณุโลก 65000

Department of Civil Engineering, Faculty of Engineering, Naresuan University

[2]ภาควิชาฟิสิกส์ คณะวิทยาศาสตร์ มหาวิทยาลัยนเรศวร จังหวัดพิษณุโลก 65000

Department of Physics, Faculty of Science, Naresuan University, Phisanulok, 65000

Abstract

The purpose of this research is to design and develop an intelligent dust-fall jar. The main theme is to prevent the jar from the rain, during the rainy seasons the dust-fall jar technique gets affected due to the rain fall. The smart system consisting of a microcontroller, rain detection sensor, outdoor weather station sensor (light, temperature, humidity), motor controller, gear motor and off-grid solar power system. The results of the tests show as satisfactory, the reliability of the various units and the system as a whole and at the end of the design of the work. The rain sensor is activated when there is rainfall, and it will close the roof as soon as the rain is detected. Once rain is stopped, system automatically opens the roof. Moreover, the system also sends real-time data to the manager via the Blynk application.

Keyword : Internet of Things, Dust-fall jar, Intelligent

E-mail address : Chirawat8578@gmail.com