ราวตากผ้าอัตโนมัติ

AUTOMATIC CLOTHES DRYING RACK

โดย

นายพัทธวิชญ์ อาริยชัยนันต์

นายภูรี ประดิษฐบุญ

นายรัชต์พงษ์ แพทย์ประเสริฐ

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2564

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ราวตากผ้าอัตโนมัติ

AUTOMATIC CLOTHES DRYING RACK

โดย

นายพัทธวิชญ์	อาริยชัยนันต์	61010722
นายภูรี	ประดิษฐบุญ	61010840
นายรัชต์พงษ์	แพทย์ประเสริฐ	61010894

อาจารย์ที่ปรึกษา

ศ. ดร.ชูวงค์ พงศ์เจริญพาณิชย์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2564

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2564

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ราวตากผ้าอัตโนมัติ

AUTOMATIC CLOTHES DRYING RACK

ผู้จัดทำ

1.	นายพัทธวิชญ์	อาริยชัยนันต์	61010722
2.	นายภูรี	ประดิษฐบุญ	61010840
3.	นายรัชต์พงษ์	แพทย์ประเสริฐ	61010894

..... **อาจารย์ที่ปรึกษา**

(ศ. ดร.ชูวงค์ พงศ์เจริญพาณิชย์)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์เรื่อง “ราวตากผ้าอัตโนมัติ” จะไม่สามารถสำเร็จลุล่วงไปได้ด้วยดี หากไม่ได้รับความอนุเคราะห์และสนับสนุนอย่างดียิ่งจากอาจารย์ที่ปรึกษาปริญญานิพนธ์คือ ศ. ดร.ชูวงค์ พงศ์เจริญพาณิชย์ ที่กรุณาให้คำแนะนำ คำปรึกษา และแนวทางแก้ไขปัญหาที่เป็นประโยชน์ต่อการศึกษา ค้นคว้าวิจัยให้ปริญญานิพนธ์นี้สำเร็จสมบูรณ์ยิ่งขึ้น รวมถึงฝึกฝนให้ผู้จัดทำมีทักษะทางด้านการคิด การอ่าน การเขียน และการนำเสนอผลงานวิชาการ ขอขอบพระคุณความห่วงใย และความปรารถนาดีให้แก่ผู้จัดทำเป็นอย่างยิ่ง

ขอขอบคุณคณาจารย์และเจ้าหน้าที่ประจำภาควิชาวิศวกรรมโทรคมนาคม คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบังทุกท่าน ที่ได้อบรมสั่งสอนประสิทธิ์ประสาทวิชา ความรู้ และประสบการณ์ให้แก่ผู้จัดทำ

ขอขอบคุณเพื่อนร่วมงานและรุ่นพี่ทุกท่านในห้องปฏิบัติการทดลองภาควิชาวิศวกรรมโทรคมนาคม คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ที่กรุณาให้คำแนะนำ คำปรึกษา และถ่ายทอดวิชาความรู้ให้แก่ผู้จัดทำ

สุดท้ายนี้ขอกราบขอบพระคุณ บิดา มารดา และครอบครัว ที่ให้ความรัก ความห่วงใย และเป็นกำลังใจที่สำคัญเสมอมาในยามที่เกิดปัญหา จนกระทั่งปริญญานิพนธ์นี้สำเร็จลุล่วงไปได้ด้วยดีและที่สำคัญคือสนับสนุนให้โอกาสทางการศึกษาอันมีค่ายิ่งแก่ผู้จัดทำ

นายพัทธวิชญ์ อาริยชัยนันต์
นายภูรี ประดิษฐบุญ
นายรัชต์พงษ์ แพทย์ประเสริฐ
ผู้จัดทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ราวตากผ้าอัตโนมัติ

AUTOMATIC CLOTHES DRYING RACK

โดย นายพัทธวิชญ์ อาริยชัยนันต์ 61010722
นายภูรี ประดิษฐบุญ 61010840
นายรัชต์พงษ์ แพทย์ประเสริฐ 61010894

อาจารย์ที่ปรึกษา ศ. ดร.ชูวงค์ พงศ์เจริญพาณิชย์

บทคัดย่อ

ปัจจุบันแล้วผู้คนส่วนใหญ่จำเป็นที่จะต้องออกไปทำงานนอกบ้าน ในบางครั้งอาจไม่มีเวลาเพียงพอที่จะบริหารจัดการงานบ้าน การตากและเก็บผ้าจึงเป็นอีกหนึ่งงานบ้านที่ยากจะจัดการ นอกจากปัจจัยในเรื่องเวลายังมีปัจจัยทางด้านความแปรปรวนของสภาพอากาศที่ยากจะคาดเดา โครงงานนี้จึงนำเสนออุปกรณ์ที่ชื่อว่า ราวตากผ้าอัตโนมัติ อุปกรณ์ที่สามารถตากผ้า และเก็บผ้า โดยใช้ Raspberry Pi ทำงานร่วมกับเซนเซอร์วัดความชื้นและอุณหภูมิ เซนเซอร์วัดความเข้มแสง โมดูลตรวจจับน้ำฝน และมอเตอร์ โดยใช้ภาษาไพธอนในการควบคุม ราวตากผ้านี้จะทำหน้าที่เก็บผ้าเมื่อตัวต้านทานไวแสงได้รับความสว่างน้อย หรือโมดูลตรวจจับน้ำฝนมีน้ำ จะนำผ้าออกมาตากเมื่อตัวเซนเซอร์วัดความเข้มแสงได้รับความสว่างที่เพียงพอ และโมดูลตรวจจับน้ำฝนไม่มีน้ำ เพื่ออำนวยความสะดวกแก่ผู้ใช้งาน ประหยัดเวลา และสามารถทำงานได้ในสภาพอากาศที่ไม่แน่นอนได้ ทั้งในรูปแบบอัตโนมัติ และการใช้งานด้วยมือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ABSTRACT

Nowadays, most people need to work outside the home. Sometimes there may not be enough time to manage housework. Drying and storing clothes is another difficult housework to deal with. In addition to the time factor, there is also an unpredictable factor about weather variability. This project presents a device called automatic clothes drying rack. A device can dry and store clothes using Raspberry Pi that works with humidity and temperature sensors, light intensity sensor, Rain water detection module and motor used by Python language to control it. This drying rack would keep the clothes inside when the photosensitive resistor is low on brightness or the rainwater detection module has water. The drying rack would be taken out to dry. When the light intensity sensor gets enough brightness and the rainwater detection module has no water. This project is accommodated in the matter of saving time and be able to working in unpredictable weather conditions to users both in automatic and manual form.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

		หน้า
กิตติกรรมประกาศ		I
บทคัดย่อ		II
สารบัญ		IV
สารบัญรูป		IX
สารบัญตาราง		XI
บทที่ 1	**บทนำ**	**1**
	1.1 ความเป็นมาและความสำคัญของปัญหา	1
	1.2 วัตถุประสงค์	1
	1.3 ขอบเขตของปริญญานิพนธ์	2
บทที่ 2	**ทฤษฎีและหลักการที่เกี่ยวข้อง**	**3**
	2.1 หลักการที่เกี่ยวข้องกับไอโอที	4
	2.2 โมดูลขับมอเตอร์ L298N	5
	2.3 มอเตอร์ไฟฟ้า	6
	2.4 ลิมิตสวิตช์ชนิดก้านบานพับ	7
	2.5 บอร์ดไข่ปลา	8
	2.6 ตัวต้านทาน	8
	2.7 แบตเตอรี่	9
	2.8 ท่อ PVC	9
	2.9 แผ่นอะคริลิค	10
	2.10 เมทัลชีท	11
	2.11 โมดูลตรวจจับน้ำฝน	11
	2.12 เซนเซอร์วัดค่าความชื้นและอุณหภูมิ DHT22	12

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

		หน้า
	2.13 เซนเซอร์ตรวจวัดความเข้มแสง GY-2561 TSL 2561	12
	2.14 สวิตช์สองทาง	13
	2.15 Raspberry Pi	14
	2.16 Python	15
	2.17 ทฤษฎีทางกลศาสตร์	16
	2.18 ระบบปฏิบัติการ Android	18
	2.19 Adroid Studio	19
	2.20 JAVA	20
	2.21 Firebase	21
บทที่ 3	**การออกแบบและการจัดทำปริญญานิพนธ์**	22
	3.1 การออกแบบ	22
	3.2 เครื่องมือที่ใช้ในการทดลอง	38
	3.3 การจัดเก็บผลการทดลอง	40
บทที่ 4	**ผลการทดลอง**	47
	4.1 ผลการทดสอบการทำงานของเซนเซอร์ตรวจวัดความเข้มแสง	47
	4.2 ผลการทดสอบการทำงานของโมดูลตรวจจับน้ำฝน	49
	4.3 ผลการทำสอบการทำงานของเซนเซอร์วัดความชื้นและอุณหภูมิ	50
	4.4 ผลการทดสอบการทำงานมอเตอร์	50
	4.5 ผลการทดสอบการทำงานตามสภาพภูมิอากาศ	51
	4.6 ผลการทดสอบการเลื่อนของรอก	51
	4.7 ผลการทดสอบการรับส่งข้อมูลระหว่าง Raspberry Pi กับฐานข้อมูล Firebase Realtime Database	52

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

		หน้า
	4.8 ผลการทดสอบการเชื่อมต่อระหว่างฐานข้อมูล Firebase Realtime Database กับ แอปพลิเคชัน	53
	4.9 ผลการทดสอบการใช้งานแอปพลิเคชัน	54
บทที่ 5	**สรุปผลและข้อเสนอแนะ**	**58**
	5.1 สรุปผล	58
	5.2 ข้อเสนอแนะ	58
บรรณานุกรม		**59**
ภาคผนวก ก	ชุดคำสั่งการรับข้อมูลจากเซนเซอร์บน Raspberry Pi	62
ภาคผนวก ข	ชุดคำสั่งที่ใช้ส่งข้อมูลเซนเซอร์จาก Raspberry Pi ไปยัง Firebase Realtime Database	71
ภาคผนวก ค	ชุดคำสั่งที่ใช้ควบคุมการทำงานของมอเตอร์ตามเงื่อนไขทั้งหมด	78
ภาคผนวก ง	ชุดคำสังที่ใช้เริ่มการทำงานอัตโนมัติเมื่อเปิดใช้งาน	84
ภาคผนวก จ	ชุดคำสั่งการเปลี่ยนหน้าบนแอปพลิเคชัน	86
ภาคผนวก ฉ	ชุดคำสังที่ใช้แสดงลลัพธ์ของเซนเซอร์บนแอปพลิเคชัน	89
ภาคผนวก ช	ชุดคำสั่งที่ใช้ควบคุมการทำงานของมอเตอร์บนแอปพลิเคชัน	94

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

หน้า

รูปที่

2.1 บล็อกไดอะแกรมของโครงงาน 3
2.2 พอร์ตการเชื่อมต่อบนโมดูลขับมอเตอร์ 5
2.3 มอเตอร์ไฟฟ้า 6
2.4 หลักการทำงานของมอเตอร์ 7
2.5 ลิมิตสวิตช์ชนิดก้านบานพับ 7
2.6 บอร์ดไข่ปลา 8
2.7 ตัวต้านทาน 8
2.8 แบตเตอรี่ 9
2.9 ท่อ PVC 10
2.10 แผ่นอะคริลิค 10
2.11 เมทัลชีท 11
2.12 โมดูลตรวจจับน้ำฝน 11
2.13 เซนเซอร์วัดความชื้นและอุณหภูมิ DHT22 12
2.14 เซนเซอร์ตรวจวัดความเข้มแสง GY-2561 TSL 2561 13
2.15 สวิตช์สองทาง 14
2.16 External Port บนบอร์ด Raspberry Pi 15
2.17 หน้าต่างตัวอย่างการออกแบบหน้าจอบ โดยใช้ภาษา XML ผ่าน Android Studio 20
2.18 หน้าต่างตัวอย่างการออกแบบหน้าจอบ โดยใช้ภาษา JAJA ผ่าน Android Studio 21
3.1 โครงสร้างราวตากผ้าอัตโนมัติ 22
3.2 รอกภายในรางของรางตากผ้า 23
3.3 ภาพจำลองมุมมองด้านหน้าของราวตากผ้า 24

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

หน้า

รูปที่

3.4 โครงสร้างราวตากผ้าอัตโนมัติ 25
3.5 รอกภายในรางของรางตากผ้า 25
3.6 ลิมิตสวิตช์ภายในรางของราวตากผ้า 26
3.7 มอเตอร์ภายในราวตากผ้า 26
3.8 เซนเซอร์ตรวจวัดความเข้มแสงบริเวณด้านบนของหลังคา 27
3.9 โมดูลตรวจจับน้ำฝนบริเวณด้านบนของหลังคา 27
3.10 เซนเซอร์วัดความชื้นและอุณหภูมิบริเวณชายคาของหลังคา 28
3.11 การเลื่อนรอกเข้า 28
3.12 การเลื่อนรอกออก 29
3.13 การเชื่อมต่ออุปกรณ์ต่าง ๆ 30
3.14 แผนผังหลักการทำงานของราวตากผ้าอัตโนมัติ 31
3.15 ชื่อที่อยู่ของไฟล์ที่เชื่อมระหว่างอุปกรณ์ และ Firebase 34
3.16 หน้าแรกของแอปพลิเคชัน 35
3.17 หน้า DISPLAY ของแอปพลิเคชัน 36
3.18 ตัวอย่างการควบคุมโหมดการทำงานของราวตากผ้าอัตโนมัติ 37
3.19 ตัวอย่างโหมดการควบคุมการจัดเก็บของราวตากผ้าอัตโนมัติ 38
3.20 Lenovo IdeaPad Gaming 39
3.21 Realme 6 Pro 40
3.22 แผนภาพการเชื่อมต่อเซนเซอร์วัดความเข้มแสงกับ Raspberry Pi 41
3.23 แผนภาพการเชื่อมต่อเซนเซอร์โมดูลตรวจจับน้ำฝนกับ Raspberry Pi 42
3.24 แผนภาพการเชื่อมต่อเซนเซอร์วัดความชื้นและอุณหภูมิกับ Raspberry Pi 43
3.25 แผนภาพการเชื่อมต่อมอเตอร์และโมดูลขับมอเตอร์กับ Raspberry Pi 44
3.26 ราวตากผ้าเมื่อรอกเคลื่อนที่เข้า 45
3.27 ราวตากผ้าเมื่อรอกเคลื่อนที่ออก 45

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

หน้า

รูปที่

3.28 ราวตากผ้าขณะตากเมื่อรอกเคลื่อนที่ออก 45

3.29 ราวตากผ้าขณะตากเมื่อรอกเคลื่อนที่เข้า 46

4.1 ความเข้มแสงในช่วงเวลา 6 นาฬิกา ถึง 18 นาฬิกา 48

4.2 ผลการทดสอบเมื่อโมดูลตรวจจับน้ำฝนตรวจไม่พบน้ำ 49

4.3 ผลการทดสอบเมื่อโมดูลตรวจจับน้ำฝนตรวจพบน้ำ 49

4.4 ผลการทดสอบเซนเซอร์วัดความชื้นและอุณหภูมิ 50

4.5 ผลการทำสอบชื่อที่อยู่ของไฟล์ที่เชื่อมต่อระหว่างอุปกรณ์ และ Firebase Realtime Database 52

4.6 ผลการทดสอบการเชื่อมต่อฐานข้อมูล Firebase Realtime Database กับแอปพลิเคชันในหน้า Display 53

4.7 ผลการทดสอบการเชื่อมต่อฐานข้อมูล Firebase Realtime Database กับแอปพลิเคชันในหน้า Control 54

4.8 ผลการควบคุมรอกให้อยู่ในตำแหน่งเก็บผ้า (inside) 57

4.9 ผลการควบคุมรอกให้อยู่ในตำแหน่งตากผ้า (outside) 57

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

หน้า

ตารางที่

4.1 ค่าความเข้มแสงในทุกๆ 1 ชั่วโมงตั้งแต่เวลา 6 นาฬิกา ถึงเวลา 18 นาฬิกา เป็นเวลา 3 วัน 47

4.2 ตารางทดสอบการหมุนของมอเตอร์ 50

4.3 ผลการทดสอบการหมุนของมอเตอร์ 51

4.4 เปรียบเทียบเวลาการเลื่อนของรอกเมื่อไม่มีผ้า และมีผ้าเปียก 52

4.5 การแสดงผลข้อมูลของเซนเซอร์ทุกๆ 1 ชั่วโมงตั้งแต่เวลา 6 นาฬิกา ถึงเวลา 18 นาฬิกาในวันที่ 1 55

4.6 การแสดงผลข้อมูลของเซนเซอร์ทุกๆ 1 ชั่วโมงตั้งแต่เวลา 6 นาฬิกา ถึงเวลา 18 นาฬิกาในวันที่ 2 55

4.7 การแสดงผลข้อมูลของเซนเซอร์ทุกๆ 1 ชั่วโมงตั้งแต่เวลา 6 นาฬิกา ถึงเวลา 18 นาฬิกาในวันที่ 3 56

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ปัจจุบันสภาพภูมิอากาศเป็นสิ่งที่มนุษย์ไม่สามารถควบคุมได้ ไม่ว่าจะเป็นแสงแดด อุณหภูมิ ความชื้น หรือน้ำฝน ซึ่งสิ่งเหล่านี้เป็นอุปสรรคในการทำให้เสื้อผ้าแห้ง ถึงแม้ว่าปัจจุบันมีการใช้เครื่องอบผ้ากันอย่างแพร่หลาย แต่ด้วยราคาที่สูง โครงงานนี้จึงเป็นอีกหนึ่งทางเลือกเพื่ออำนวยความสะดวกแก่ผู้ใช้งาน ผนวกกับสังคมในปัจจุบันที่ผู้คนส่วนใหญ่นั้นออกไปทำงานนอกบ้าน หากสภาพภูมิอากาศไม่เอื้ออำนวยต่อการตากผ้าในขณะที่ไม่ได้อยู่ที่บ้าน อาจะทำให้เสื้อผ้าเปียกและอับชื้น ส่งผลให้ต้องกลับมาซักผ้าซ้ำอีกครั้ง จากปัญหาดังกล่าวจึงเป็นเหตุให้ผู้จัดทำเสนอปริญญานิพนธ์โดยนำเทคโนโลยีไอโอทีซึ่งมีบทบาทอย่างแพร่หลายในอุตสาหกรรมต่าง ๆ มาประยุกต์ใช้ร่วมกับไมโครคอนโทรลเลอร์เพื่อแก้ไขปัญหาข้างต้น

ราวตากผ้าอัตโนมัติได้รับการออกแบบโดยใช้ Raspberry Pi ทำงานร่วมกับ เซนเซอร์และมอเตอร์ซึ่งออกแบบการทำงานด้วยภาษาไพธอนในการควบคุมการทำงานของชิ้นงาน เพื่อให้สามารถตากและจัดเก็บผ้าได้ตามสภาพแวดล้อมที่ได้ทำการกำหนดเงื่อนไขไว้ ในรูปแบบอัตโนมัติและระบบสั่งการด้วยมือ พร้อมทั้งมีการแสดงผลผ่านแอปพลิเคชันบนสมาร์ทโฟน

1.2 วัตถุประสงค์

1) สร้างราวตากผ้าที่สามารถตากและเก็บผ้าโดยใช้ระบบอัตโนมัติจากการรับค่าเซนเซอร์มาประมวลผลและระบบสั่งการด้วยมือได้
2) เพื่ออำนวยความสะดวกสบายให้กับผู้ใช้งานโดยสามารถติดตั้งราวตากผ้าอัตโนมัติในพื้นที่ที่มีสภาพอากาศไม่แน่นอนได้
3) เพื่อศึกษาทฤษฎีและหลักการทำงานของอุปกรณ์ต่าง ๆ โดยเป็นการนำองค์ความรู้ที่ได้มาประยุกต์ใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3 ขอบเขตของปริญญานิพนธ์

ขอบเขตของปริญญานิพนธ์นี้เป็นการใช้เทคโนโลยีไอโอทีมาประยุกต์ใช้กับราวตากผ้าให้สามารถตากและจัดเก็บผ้าเข้าในที่พักได้โดยอัตโนมัติโดยมีระบบตรวจจับแสง อุณหภูมิ และน้ำฝน โดยใช้เซนเซอร์เพื่อควบคุมมอเตอร์ผ่าน Raspberry Pi มีระบบสั่งการด้วยมือและตรวจสอบสภาพอากาศได้ผ่านทางแอปพลิเคชัน เพื่อให้ผู้ใช้งานไม่จำเป็นต้องสังเกตสภาพภูมิอากาศตลอดเวลาเมื่อทำการตากผ้า หรือสามารถควบคุมการตากผ้าได้จากระยะไกล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

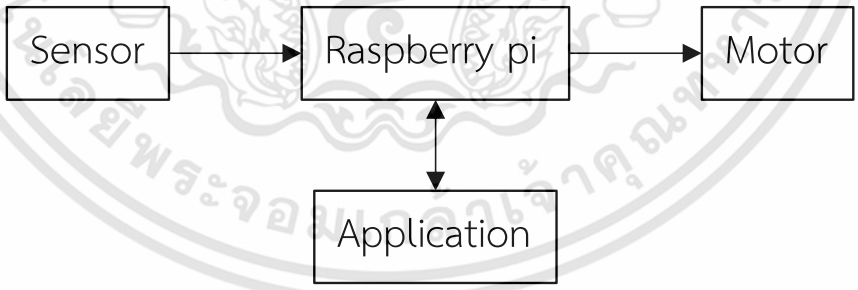
บทที่ 2

ทฤษฎีและหลักการที่เกี่ยวข้อง

ปริญญานิพนธ์นี้เป็นการออกแบบราวตากผ้าอัตโนมัติที่สามารถตากและเก็บผ้าได้ด้วยระบบอัตโนมัติจากเซนเซอร์โดยเป็นการประยุกต์เทคโนโลยีไอโอที (Internet of Things: IoT) และองค์ความรู้ของวิศวกรรม เพื่อการออกแบบให้ได้มาซึ่งราวตากผ้าอัตโนมัติที่มีประสิทธิภาพและสามารถทำงานได้อย่างเหมาะสม

ชิ้นงานดังกล่าวประกอบด้วยเซนเซอร์วัดความเข้มแสง เซนเซอร์ตรวจจับน้ำฝน และเซนเซอร์วัดความชื้นและอุณหภูมิ โดยเซนเซอร์ต่าง ๆ จะติดตั้งบริเวณใต้หลังคาชิ้นงาน โดยจะทำงานร่วมกับ Raspberry Pi เพื่อทำหน้าที่ตรวจสอบน้ำ แสงแดด อุณหภูมิ และความชื้น เพื่อสั่งการมอเตอร์พร้อมแสดงผลไปยังแอปพลิเคชัน โดยใช้ภาษาไพธอนในการออกแบบคำสั่งในการควบคุมการทำงานของชิ้นงาน

หลักการทำงานของชิ้นงานเริ่มจากส่วนของเซนเซอร์ทำการตรวจสอบสภาพแวดล้อมตามเงื่อนไขที่กำหนด โดยทำการส่งค่าที่รับได้ไปยัง Raspberry Pi เพื่อประมวลผลและควบคุมการทำงานของมอเตอร์ที่ติดตั้งไว้กับชิ้นงาน และบันทึกข้อมูลส่งไปยังแอปพลิเคชัน ซึ่งสามารถใช้งานระบบสั่งการด้วยมือและตรวจสอบสภาพอากาศได้ผ่านทางแอปพลิเคชัน สามารถแสดงการทำงานด้วยบล็อกไดอะแกรมแสดงดังรูปที่ 2.1



รูปที่ 2.1 บล็อกไดอะแกรมของโครงงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1 หลักการที่เกี่ยวข้องกับไอโอที

2.1.1 เทคโนโลยีไอโอที

ปัจจุบันอินเทอร์เน็ตเป็นการเชื่อมต่อผู้คนในโลกมากกว่าหนึ่งพันล้านให้สื่อสารกันได้ผ่านอีเมล โซเชียลมีเดีย หรือแชท แต่การเชื่อมต่อไม่ได้หยุดเพียงเท่านั้น นักพัฒนาเริ่มมองหา และสร้างสิ่งที่เรียกว่าไอโอทีให้เป็นจริงขึ้นมา ไอโอที คือ เทคโนโลยีอินเทอร์เน็ตที่เชื่อมอุปกรณ์ให้สื่อสารกันได้ อาทิ สมาร์ทโฟน เครื่องคอมพิวเตอร์ รถยนต์ ตู้เย็น เครื่องซักผ้า โทรทัศน์ กล้องวงจรปิด หรือเซนเซอร์ต่าง ๆ เข้าด้วยกัน โดยผ่านเครือข่ายอินเทอร์เน็ต ในอนาคตมนุษย์จะคุ้นชินกับการควบคุมสิ่งรอบตัวที่อยู่ในระยะใกล้และไกล และเรียกทุกอย่างที่เชื่อมต่อกันเป็นไอโอทีว่า ระบบอัจฉริยะ เพราะสามารถทำงานได้โดยไม่ต้องรอรับการสั่งงาน

ตัวอย่างการควบคุมที่ใกล้เคียงกับการใช้งานในชีวิตประจำวัน อาทิ การควบคุมอุณหภูมิภายในบ้านให้เหมาะกับการอยู่อาศัย การเปิดปิดไฟอัตโนมัติ และการใช้อุปกรณ์อัตโนมัติทำงานแทนคน

อุปกรณ์สำคัญที่สนับสนุนให้ไอโอทีนำมาใช้งานได้จริง คือ เซนเซอร์ที่ถูกฝังตัวเข้ากับเครื่องใช้ไฟฟ้ารอบตัวเรา ซึ่งสามารถเก็บบันทึกข้อมูล รับรู้สภาพแวดล้อม สามารถควบคุมได้จากระยะไกล และสามารถประยุกต์ใช้กับอุปกรณ์ทุกอย่างที่เชื่อมต่อกับเครือข่ายอินเทอร์เน็ตเพื่อที่จะสื่อสารถึงกันได้ [1]

2.1.2 องค์ประกอบของไอโอที

ทุกองค์ประกอบในการทำงานของไอโอทีจะต้องทำงานสอดประสานกัน เพื่อให้ระบบทำหน้าที่ได้ลุล่วงและทำงานได้อย่างอัตโนมัติ ผู้ใช้มีหน้าที่เพียงติดตั้ง ซ่อมแซม และรับข้อมูลเท่านั้น โดยองค์ประกอบทั้งหมดจะมีทั้งหมด 4 ส่วน ดังนี้

1.เซนเซอร์ ใช้เป็นหน่วยรับข้อมูลที่มีความสามารถในการตรวจวัดสถานะในบริเวณที่สนใจ อาจมีการเชื่อมต่อกับกลไกควบคุมระบบไฟฟ้าเพื่อสั่งงาน

2.อุปกรณ์เกตเวย์ (Gateway) คืออุปกรณ์สำหรับการเชื่อมต่ออุปกรณ์ไอโอทีกับโครงข่ายอินเทอร์เน็ต อาจเป็นได้ทั้งเครือข่ายภายในหรือเครือข่ายสาธารณะ

3.เครื่องบริการหรือเครื่องเซิร์ฟเวอร์ (Server) เป็นเครื่องคอมพิวเตอร์ที่ทำหน้าที่เป็นจุดศูนย์รวม ประมวลผล หรือเชื่อมโยงข้อมูล แล้วส่งต่อไปยังอุปกรณ์อื่นที่ต้องการข้อมูล

4.อุปกรณ์ฝั่งผู้ใช้ (User Device) เป็นส่วนการแสดงผลสถานะที่ตรวจวัดได้จากเซนเซอร์ให้ผู้ใช้รับทราบข้อมูลในรูปของแอปพลิเคชัน [2]

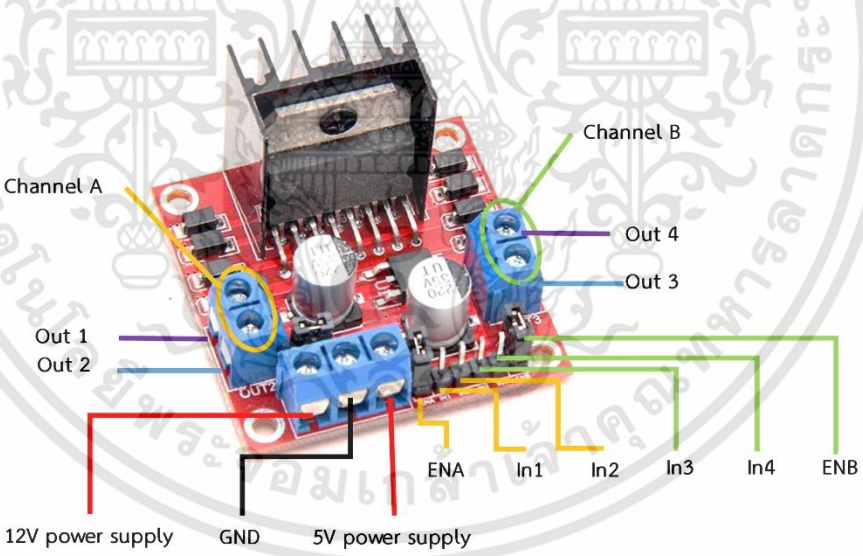
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 โมดูลขับมอเตอร์ L298N

โมดูลขับมอเตอร์ L298N (L298N motor drive) เป็นชุดขับมอเตอร์ชนิด H-Bridge ส่วนใหญ่ถูกนำไปใช้ควบคุมทิศทาง และความเร็วของมอเตอร์ ซึ่งสามารถควบคุมมอเตอร์ได้ 2 Channel ซึ่งวงจร H-Bridge ของ L298N จะขับกระแสไฟฟ้าเข้ามอเตอร์ตามขั้วที่กำหนดด้วยลอจิก เพื่อควบคุมทิศทาง โดยความเร็วของมอเตอร์จะควบคุมด้วยการมอดูเลตความกว้างพัลส์ (Pulse-width Modulation: PWM)

PWM คือการส่งสัญญาณแบบสวิตช์ หรือ ส่งค่าดิจิทัล 0 ถึง 1 การควบคุมระยะเวลาสัญญาณสูงและสัญญาณต่ำที่ต่างกัน ทำให้ค่าแรงดันไฟฟ้าเฉลี่ยของสัญญาณสวิตช์ต่างกัน

L298N motor drive แบ่ง Out 1 และ Out 2 เป็นช่องต่อขั้วไฟฟ้าของมอเตอร์ Channel 1 Out 3 และ Out 4 เป็นช่องต่อขั้วไฟฟ้าของมอเตอร์ Channel 2 มีช่องแรงดันไฟฟ้าเลี้ยงมอเตอร์ขนาด 12 โวลต์ 5 โวลต์ และ GND มี 6 Pin ENA และ ENB สำหรับเป็นพอร์ต PWM ในแต่ละ Channel สำหรับ In1 In2 เป็นพอร์ตที่เป็นลอจิกเพื่อควบคุมทิศทางของ Channel A และ In3 In4 เป็นพอร์ตที่เป็นลอจิกเพื่อควบคุมทิศทางของ Channel B แสดงดังรูปที่ 2.2 [3]



รูปที่ 2.2 พอร์ตการเชื่อมต่อบนโมดูลขับมอเตอร์ [3]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 มอเตอร์ไฟฟ้า

มอเตอร์ไฟฟ้า (Electric motor) เป็นอุปกรณ์ไฟฟ้าที่แปลงพลังงานไฟฟ้าเป็นพลังงานกล การทำงานปกติของมอเตอร์ไฟฟ้าส่วนใหญ่เกิดจากการทำงานร่วมกันระหว่างสนามแม่เหล็กของแม่เหล็กในตัวมอเตอร์ และสนามแม่เหล็กที่เกิดจากกระแสไฟฟ้าภายในขดลวดทำให้เกิดแรงดูดและแรงผลักของสนามแม่เหล็กทั้งสอง ในการใช้งานตัวอย่างเช่น ในอุตสาหกรรมการขนส่งใช้มอเตอร์ฉุดลาก เป็นต้น นอกจากนั้นแล้ว มอเตอร์ไฟฟ้า แสดงดังรูปที่ 2.3 ยังสามารถทำงานได้ถึงสองแบบได้แก่ การสร้างพลังงานกล และการผลิตพลังงานไฟฟ้า มอเตอร์ไฟฟ้าถูกนำไปใช้งานที่หลากหลายเช่น พัดลมอุตสาหกรรม เครื่องเป่า ปั๊ม และเครื่องมือเครื่องใช้ในครัวเรือน มอเตอร์ไฟฟ้าสามารถขับเคลื่อนโดยแหล่งจ่ายไฟฟ้ากระแสตรงเช่น แบตเตอรี่ หรือแหล่งจ่ายไฟฟ้ากระแสสลับ เช่น ไฟบ้านหรือเครื่องปั่นไฟ เป็นต้น [4]

รูปที่ 2.3 มอเตอร์ไฟฟ้า [4]

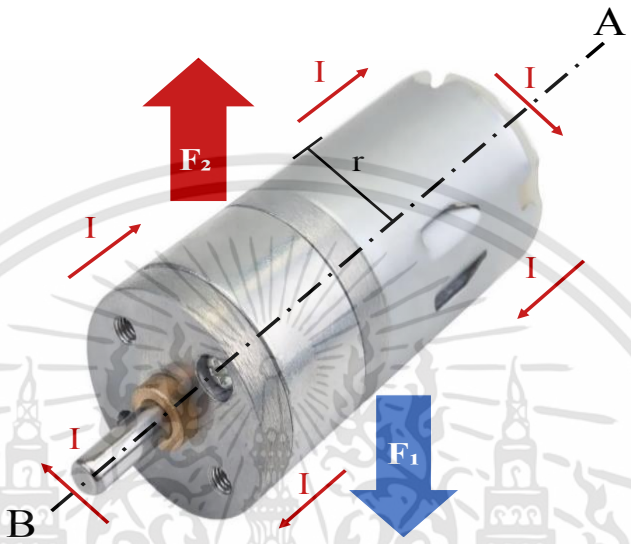
2.3.1 หลักการทำงานของมอเตอร์

การทำงานของมอเตอร์ไฟฟ้าส่วนใหญ่เกิดจากการทำงานร่วมกันระหว่างสนามแม่เหล็กของแม่เหล็กในตัวมอเตอร์ และสนามแม่เหล็กที่เกิดจากกระแสไฟฟ้าในขดลวดตัวนำทำให้เกิดแรงดูดและแรงผลักของสนามแม่เหล็กทั้งสอง

เมื่อกระแสไฟฟ้าไหลผ่านขดลวดตัวนำบนแกนอาร์เมเจอร์ ก่อให้เกิดแรงแม่เหล็กขึ้นรอบขดลวดตัวนำ และทำปฏิกิริยากับเส้นแรงแม่เหล็กที่เกิดจากขั้วแม่เหล็กของมอเตอร์ ทำให้เกิดแรงผลักขึ้นบนขดลวดตัวนำ ทำให้แกนอาร์เมเจอร์หมุนได้ ขดลวดที่มีกระแสไฟฟ้าไหลผ่านและวางอยู่บนแกนของอาร์เมเจอร์ โดยวางห่างจากจุดศูนย์กลางเป็นระยะ r กำหนดให้กระแสไฟฟ้าไหลเข้าขดลวดที่ปลาย A และไหลออกที่ปลาย B จากคุณสมบัติของเส้นแรงแม่เหล็กจะไม่ตัดผ่านซึ่งกันและกัน ดังนั้นปริมาณของเส้นแรงแม่เหล็กจะมีจำนวนมากที่ด้านบนของปลาย A จึงทำให้เกิดแรง F1 กดตัวนำ A ลงด้านล่างและขณะเดียวกันที่ปลาย B นั้น เส้นแรงแม่เหล็กจะมีปริมาณมากที่ด้านหน้าทำให้เกิดแรง F2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลักให้ตัวนำ B เคลื่อนที่ด้านบนของแรง F1 และ F2 นี้เองทำให้อาร์เมเจอร์ของมอเตอร์เกิดการเคลื่อนที่ได้ แสดงดังรูปที่ 2.4 [5]



รูปที่ 2.4 หลักการทำงานของมอเตอร์ [5]

2.4 ลิมิตสวิตช์ชนิดก้านบานพับ

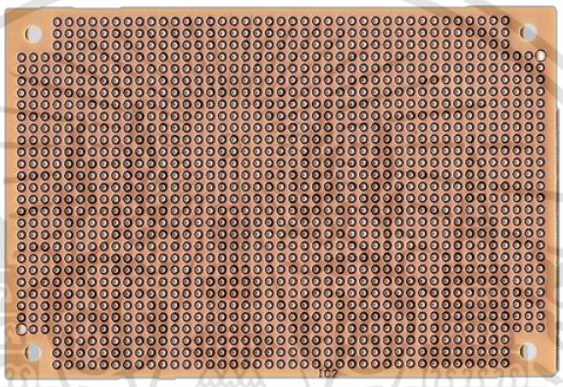
ลิมิตสวิตช์ชนิดก้านบานพับ (Hinge Lever Type) แสดงดังรูปที่ 2.5 ประกอบด้วยก้านบานพับแบนเรียบอยู่ที่ด้านข้าง ซึ่งมีทั้งแบบมีหัวลูกล้อและไม่มีหัวลูกล้อ ส่วนกลไกการเปิดปิดสวิตช์มีทั้งแบบกดและแบบสปริงให้เลือกใช้งาน โดยการทำงานอาศัยแรงจากภายนอกในทิศทางเดียว เมื่อก้านบานพับส่งแรงกดทับที่หน้าสัมผัสจะทำให้เกิดสภาวะการทำงาน จึงเหมาะสำหรับใช้งานที่ต้องการความแข็งแรงเชิงกลสูง [6]

รูปที่ 2.5 ลิมิตสวิตช์ชนิดก้านบานพับ [6]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5 บอร์ดไข่ปลา

บอร์ดไข่ปลา (Perfboard) คือ แผ่นวงจรที่มีรูเจาะไว้ทั่วทั้งแผ่นอย่างสม่ำเสมอในแนวตารางสี่เหลี่ยมแสดงดังรูปที่ 2.6 โดยทั่วไปจะห่างกันรูละ 2.54 มิลลิเมตร โดยบนแผ่นรอบแต่ละรูจะมีลายทองแดงเป็นรูปวงกลมหรือสี่เหลี่ยม ที่แต่ละรูจะไม่เชื่อมต่อกัน ซึ่งเมื่อนำมาใช้ประกอบวงจรจะต้องโยงสายเชื่อมต่อกับแต่ละจุดของวงจรเอง จากลายทองแดงของแผ่นวงจรพิมพ์ชนิดนี้ ที่โดยปกติจะเป็นรูปวงกลมและอยู่กันเป็นแผ่น จึงเป็นที่มาของชื่อเรียกบอร์ดไข่ปลา [7]



รูปที่ 2.6 บอร์ดไข่ปลา [7]

2.6 ตัวต้านทาน

เป็นอุปกรณ์ไฟฟ้าชนิดหนึ่งที่มีคุณสมบัติในการต้านทานการไหลผ่านของกระแสไฟฟ้า ทำด้วยลวดต้านทานหรือถ่านคาร์บอนเป็นต้น โดยปริมาณกระแสไฟฟ้าจะแปรผกผันกับค่าความต้านทาน เป็นอุปกรณ์ไฟฟ้าชนิดแบบเฉื่อยงาน (Passive Element) สองขั้ว ดังนั้นตัวต้านทานจึงเป็นอุปกรณ์ที่มีการใช้กันมากในงานด้านไฟฟ้าอิเล็กทรอนิกส์แสดงดังรูปที่ 2.7 [8]

รูปที่ 2.7 ตัวต้านทาน [8]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7 แบตเตอรี่

แบตเตอรี่ (Battery) คือ อุปกรณ์ที่ทำหน้าที่จัดเก็บพลังงาน ถือเป็นอุปกรณ์ที่สามารถแปลงพลังงานเคมี ให้เป็นพลังงานไฟฟ้าได้โดยตรงด้วยการใช้เซลล์กัลวานิก (Galvanic cell) ที่ประกอบด้วยขั้วบวกและขั้วลบ พร้อมกับสารละลาย อิเล็กโตรไลต์ (Electrolyte solution) แบตเตอรี่อาจประกอบด้วยเซลล์กัลวานิกเพียง 1 เซลล์ หรือมากกว่า

แบตเตอรี่ดังรูปที่ 2.8 เป็นอุปกรณ์สำหรับจัดเก็บไฟฟ้าเท่านั้น ไม่ได้ผลิตไฟฟ้า มีแรงดันไฟฟ้า 12 โวลต์ กระแสไฟฟ้าสูงสุด 2.16 แอมแปร์ สามารถประจุไฟฟ้าเข้าไปใหม่ (Recharge) ได้หลายครั้งแต่ประสิทธิภาพจะลดลงประมาณร้อยละ 20 เนื่องจากมีการสูญเสียพลังงานบางส่วนไปในรูปความร้อนและปฏิกิริยาเคมีจากการประจุ แบตเตอรี่จัดเป็นอุปกรณ์ที่มีราคาแพงและเสียหายได้ง่าย หากดูแลรักษาไม่ดีเพียงพอหรือใช้งานผิดวิธีรวมถึงอายุการใช้งานของแบตเตอรี่แต่ละชนิดจะแตกต่างกันไปเนื่องด้วยวิธีการใช้ การบำรุงรักษา การประจุและอุณหภูมิ เป็นต้น [9]

รูปที่ 2.8 แบตเตอรี่ [9]

2.8 ท่อ PVC

ท่อ PVC แสดงดังรูปที่ 2.9 คือ ท่อที่ทำขึ้นจากโพลิไวนิลคลอไรด์โดยไม่ผสมพลาสติกไซเซอร์ซึ่งชื่ออย่างเป็นทางการที่ได้ระบุใน มอก. คือ ท่อ PVC แข็ง แต่คนทั่วไปนั้นจะรู้จักกันในชื่อท่อ PVC โดยในปัจจุบันท่อชนิดนี้เป็นที่นิยมอย่างมากในวงการก่อสร้าง เพราะคุณสมบัติที่ดีหลายประการ ไม่ว่าจะเป็นคุณสมบัติที่มีความเหนียวยืดหยุ่นตัวได้ดี ทนต่อแรงดันน้ำ ทนต่อการกัดกร่อน ไม่เป็นฉนวนนำไฟฟ้าเพราะไม่เป็นตัวนำไฟฟ้า เป็นวัสดุไม่ติดไฟ น้ำหนักเบา และราคาถูก ถูกนำมาใช้ในงานหลายระบบ เช่น ระบบประปา และระบบงานระบายน้ำทางการเกษตรและอุตสาหกรรม เป็นต้น [10]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.9 ท่อ PVC [10]

2.9 แผ่นอะคริลิค

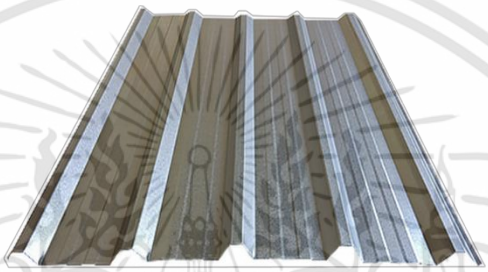
แผ่นอะคริลิค เป็นพลาสติกที่มีความทนทานและได้รับความนิยมในการนำมาใช้งานอย่างแพร่หลาย ซึ่งอะคริลิคมักจะมีหลายชื่อด้วยกัน โดยนิยมเรียกว่า อะคริลิคพลาสติก กระจกอะคริลิคหรือแผ่นอะคริลิค สำหรับคุณสมบัติของอะคริลิคเป็นวัสดุที่มีความทนทานแข็งแรง สามารถทนต่อแรงกระแทกได้ดีกว่ากระจกทั้งมีความหนาตั้งแต่ 2 มิลลิเมตร ถึง 100 มิลลิเมตรขึ้นไป จึงสามารถนำมาใช้งานได้อย่างหลากหลาย ตัวอย่างชิ้นงานจากแผ่นอะคริลิค แสดงได้ดังรูป 2.10 เช่น กรอบรูป ป้ายโฆษณาและชั้นวาง เป็นต้น [11]

รูปที่ 2.10 แผ่นอะคริลิค [11]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.10 เมทัลชีท

เมทัลชีท (Metal Sheet) คือ แผ่นเหล็กรีดลอนโลหะผสมระหว่าง อลูมิเนียม และสังกะสีแสดงดังรูปที่ 2.11 เหมาะสำหรับทั้งงานภายในและภายนอก งานผนัง งานรั้วและงานหลังคา คุณสมบัติเด่นชัดของเมทัลชีทคือสามารถสั่งผลิตตามขนาดความยาวของหลังคาได้ จึงทำให้เกิดรอยต่อของแผ่นหลังคาน้อย ปัญหารั่วซึมจึงน้อยกว่าหลังคากระเบื้องทั่วไป อีกทั้งยังช่วยให้การติดตั้งหลังคาสามารถดำเนินการได้ไว น้ำหนักเบา จึงช่วยลดต้นทุนค่าแรงและค่าโครงสร้างบ้านได้เป็นอย่างดี [12]



รูปที่ 2.11 เมทัลชีท [12]

2.11 โมดูลตรวจจับน้ำฝน

โมดูลตรวจจับน้ำฝน (Raindrop Detection Sensor Module) เป็นโมดูลสำหรับตรวจสอบสภาพอากาศ วัดความชื้นในอากาศ และน้ำฝน แสดงดังรูปที่ 2.12 ขึ้นอยู่กับการเขียนโปรแกรมและการใช้งาน สามารถเชื่อมต่อใช้งานกับ Raspberry Pi Arduino ARM MCS-51 AVR PIC แรงดันไฟฟ้าแหล่งจ่ายขนาด 3.3 โวลต์ ถึง 5 โวลต์ โมดูลขนาด 1.5 × 3.2 × 1.2 เซนติเมตร โพรบตรวจจับน้ำฝน ขนาด 4.0 × 5.5 × 1.0 เซนติเมตร [13]

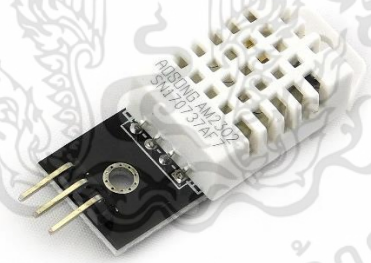
รูปที่ 2.12 โมดูลตรวจจับน้ำฝน [13]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในสภาพอากาศปกติ โมดูลมีค่าความต้านทานสูง แต่เมื่อสภาพอากาศเปลี่ยนแปลง เช่น ฝนตกหรือมีความชื้นในอากาศมากจะให้ค่าความต้านทานลดต่ำลง สามารถปรับค่าความไวในการตรวจจับได้ (Potentiometer Sensitivity Adjustment) ซึ่งมีสัญญาณขาออกในรูปแบบสัญญาณดิจิทัล คือ 0 กับ 1 หรือสัญญาณขาออกในรูปแบบแอนะล็อก และมี LED แสดงสถานะการทำงานและสัญญาณขาออก

2.12 เซนเซอร์วัดความชื้นและอุณหภูมิ DHT22

เซนเซอร์วัดความชื้นและอุณหภูมิ DHT22 (DHT22 Temperature and Humidity Sensor Module) แสดงดังรูปที่ 2.13 สามารถวัดความชื้นและอุณหภูมิบริเวณทั่วไปหรือประยุกต์ใช้งานด้านอื่นได้อย่างหลากหลาย ขึ้นอยู่กับการสั่งการ การใช้งาน การตอบสนองที่รวดเร็ว และความแม่นยำ สามารถใช้งานร่วมกับไมโครคอนโทรลเลอร์ทั่วไป โมดูลประกอบไปด้วย แรงดันไฟฟ้าแหล่งจ่าย (VCC) ขนาด 3.3 โวลต์ ถึง 5 โวลต์ โมดูลขนาด 15.5 × 12 × 5.5 มิลลิเมตร ส่วนวัดความชื้นแบบรีซีสทีฟ และส่วนวัดอุณหภูมิแบบหัววัดอุณหภูมิชนิดลบ (NTC) ให้สัญญาณขาออกแบบดิจิทัล การตรวจวัดคงที่กับ DHT22 Sensor ย่านวัดอุณหภูมิ 0 องศาเซลเซียส ถึง 50 องศาเซลเซียส ย่านวัดความชื้นสัมพัทธ์ร้อยละ 20 ถึง ร้อยละ 90 โดยมีความผิดพลาดในการวัด ±2 องศาเซลเซียส สามารถอ่านค่าได้ 1 ครั้งต่อวินาที [14]



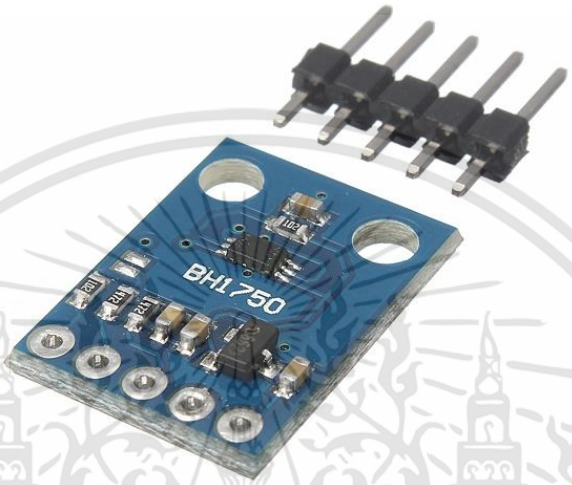
รูปที่ 2.13 เซนเซอร์วัดความชื้นและอุณหภูมิ DHT22 [14]

2.13 เซนเซอร์ตรวจวัดความเข้มแสง GY-2561 TSL 2561

TSL 2561 แสดงดังรูปที่ 2.14 เป็นโมดูลเซนเซอร์ความเข้มแสงในช่วงกว้าง เหมาะสำหรับการใช้งานในสถานการณ์ที่ความเข้มแสงมีความหลากหลายเป็นอย่างยิ่ง มีความแม่นยำสูง สามารถคำนวณความสว่างได้ตั้งแต่ช่วง 0.1 ลักซ์ ถึง 40,000 ลักซ์ ได้อย่างแม่นยำ ส่วนที่สำคัญที่สุดคือเซนเซอร์นี้สามารถคำนวณอินฟราเรดและสเปกตรัมทั้งหมดได้ หมายความว่าสามารถแยกการวัดอินฟราเรดเต็มสเปกตรัม หรือแสงที่มนุษย์มองเห็นได้ ซึ่งเซนเซอร์วัดแสงชนิดอื่นสามารถตรวจจับได้เพียงอย่างใดอย่างหนึ่งเท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เซนเซอร์นี้ใช้แรงดันไฟฟ้าขนาด 2.7 โวลต์ ถึง 3.6 โวลต์ มีการเชื่อมต่อแบบอนุกรมความเร็วต่ำ (I²C) ทำให้ สามารถนำไปใช้งานต่อได้ง่าย และยังทนความร้อนได้สูงสุดถึง 80 องศาเซลเซียส [15]



รูปที่ 2.14 เซนเซอร์ตรวจวัดความเข้มแสง GY-2561 TSL 2561 [15]

2.13.1 การสื่อสารแบบอนุกรมความเร็วต่ำ (I²C)

Inter-Integrated Circuit: I²C คือรูปแบบการสื่อสารข้อมูลอย่างหนึ่งที่สร้างขึ้นมาเพื่อสื่อสารข้อมูลความเร็วต่ำ นิยมใช้กับไมโครคอนโทรเลอร์และอุปกรณ์ต่าง ๆ ที่เกี่ยวข้อง ถูกคิดค้นขึ้นมาในปี ค.ศ. 1982 โดย Philip semiconductor ข้อดีของการสื่อสารอนุกรมแบบ I²C คือ สามารถรับส่งข้อมูลได้หลายอุปกรณ์ในโทโพโลยีแบบบัสเดียวกัน และใช้สายสัญญาณเพียง 2 เส้นในการรับส่งข้อมูล ทำให้สามารถลดจำนวนสายสัญญาณที่ใช้ในการเชื่อมต่ออุปกรณ์ลงมาก โดยสายสัญญาณทั้ง 2 เส้น แบ่งเป็น

1. SDA (Serial Data) คือ สายสัญญาณสำหรับรับส่งข้อมูล
2. SCL (Serial Clock) คือ สายสัญญาณนาฬิกา ใช้สำหรับควบคุมสัญญาณนาฬิกาในการรับส่งข้อมูล

2.14 สวิตช์สองทาง

สวิตช์สองทาง เป็นอุปกรณ์ไฟฟ้าชนิดหนึ่งที่ใช้ควบคุมวงจรกระแสไฟฟ้า ทำหน้าที่เปิดกระแสไฟฟ้าหรือตัดกระแสไฟฟ้าไม่ให้ไหลเข้าสู่เครื่องใช้ไฟฟ้า สวิตช์สองทางถูกออกแบบมาให้ติดตั้งได้ง่าย ใช้งานง่าย ตอบสนองได้รวดเร็ว และทนความร้อนสูง รองรับกระแสไฟฟ้าได้ 25 แอมแปร์ แสดงดังรูปที่ 2.15 [16]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.15 สวิตช์สองทาง [16]

2.15 Raspberry pi

Raspberry Pi เกิดขึ้นในปี 2549 ที่มหาวิทยาลัยเคมบริดจ์ ประเทศอังกฤษ โดยมีผู้ก่อตั้ง คือ Eben Upton Rob Mullins Jack Lang และ Alan Mycroft มีจุดมุ่งหมายที่จะให้ Raspberry Pi เป็นคอมพิวเตอร์ราคาต่ำที่สามารถศึกษาการทำงานของคอมพิวเตอร์พร้อมทั้งเขียนโปรแกรมอย่างง่ายได้ทันที

Raspberry Pi รองรับระบบปฏิบัติการลินุกซ์ (Linux Operating System) ได้หลายระบบ เช่น Raspbian Debian และ Arch Linux เป็นต้น ได้รับการเผยแพร่หลายรุ่น โดยทุกรุ่นมีหน่วยประมวลผล Broadcom เป็นหน่วยประมวลผลกลาง (CPU) และหน่วยประมวลผลกราฟิก (GPU) [17]

2.15.1 ระบบปฏิบัติการ Raspbian

Raspbian เป็นระบบปฏิบัติการสำหรับติดตั้งใช้งานบนบอร์ดขนาดเล็ก Raspberry Pi พัฒนามาจากระบบ Debian Linux เหมาะสำหรับนำมาใช้ทำการทดลองและงานวิจัยเกี่ยวกับระบบคอมพิวเตอร์แบบฝังตัว (Embedded system) โดย Raspbian มีความเหมาะสมกับการทำงานบน ARM Processor ซึ่งเป็นหน่วยประมวลผลโดยเฉพาะของ Raspberry Pi มากกว่า Linux distribution อื่น ๆ และสามารถใช้ประโยชน์จากหน่วยประมวลผลเหล่านี้ได้อย่างเต็มที่ [18]

2.15.2 บอร์ด Raspberry Pi

คุณสมบัติของบอร์ด Raspberry Pi แสดงดังรูปที่ 2.16

1. หน่วยประมวลผล ARM v8 15 กิกะเฮิรตซ์
2. RAM 8 กิกะบิต
3. Gigabit Ethernet 1 พอร์ต
4. USB 2.0 ทั้งหมด 2 พอร์ต
5. USB 3.0 ทั้งหมด 2 พอร์ต
6. HDMI ทั้งหมด 2 พอร์ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7. Display 1 พอร์ต

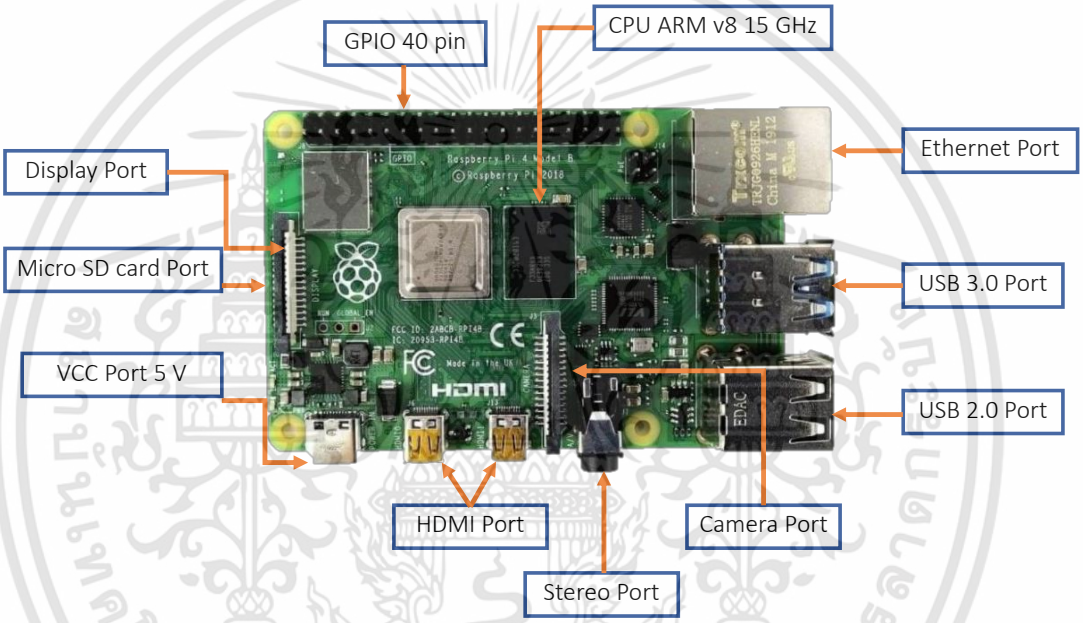
8. Camera 1พอร์ต

9. Stereo 1 พอร์ต

10. GPIO 40 pin

11. Micro SD card 1 พอร์ต

12. ช่องจ่ายแรงดันไฟฟ้ากระแสตรงขนาด 5 โวลต์ 1 พอร์ต



รูปที่ 2.16 External Port บนบอร์ด Raspberry PI

ด้วยคุณสมบัติเหล่านี้ทำให้ Raspberry Pi เหมาะสมกับการทำงานเนื่องจากสามารถรับค่าจากเซนเซอร์ เพื่อประมวลผลไปยังมอเตอร์ และแสดงผลผ่านทางแอปพลิเคชันได้อย่างมีประสิทธิภาพ

2.16 Python

ภาษาไพธอน (Python) เป็นภาษาเขียนโปรแกรมระดับสูง ที่ใช้กันอย่างกว้างขวางในการเขียน โปรแกรมสำหรับวัตถุประสงค์ทั่วไป สร้างขึ้นโดย Guido van Rossum และเผยแพร่ครั้งแรกในปี ค.ศ. 1991 ภาษาไพธอนเป็นภาษาที่ถูกออกแบบให้ชุดคำสั่งสามารถอ่านได้ง่าย มีโครงสร้างไม่ซับซ้อน โดยมักใช้คำในภาษาอังกฤษ โดย ภาษาไพธอนมีระบบการจัดการหน่วยความจำอัตโนมัติและสนับสนุนการเขียนโปรแกรมหลายรูปแบบที่ประกอบไปด้วย การเขียนโปรแกรมเชิงวัตถุ การเขียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมแบบฟังก์ชัน และการเขียนโปรแกรมแบบขั้นตอน มีไลบรารี่ที่ครอบคลุมการทำงานอย่างหลากหลาย ภาษาไพธอน มี 2 เวอร์ชัน หลัก คือ ภาษาไพธอนเวอร์ชัน 2 และภาษาไพธอนเวอร์ชัน 3 โดยภาษาไพธอนเวอร์ชัน 2 จะได้รับการสนับสนุนน้อยลงในอนาคต ส่วนภาษาไพธอนเวอร์ชัน 3 จะมีการพัฒนาต่อไปในอนาคต [19]

2.17 ทฤษฎีทางกลศาสตร์

2.17.1 การหาแรงจากน้ำหนักรวมของโหลด

จากทฤษฎีแรงโน้มถ่วง และสนามโน้มถ่วง ทุกอนุภาคสสาร โลกดึงดูดทุกอนุภาคอื่นด้วยแรงซึ่งแปรผันตรงกับผลคูณของมวลของอนุภาคและแปรผกผันกับกำลังสองของระยะห่างระหว่างอนุภาคทั้งสองนั้นดังสมการที่ (2.1) และ (2.2)

$$F = m_{\text{Total}} \times g \tag{2.1}$$

โดยสามารถหาค่า m_{Total} ได้จากสมการที่ (3.2)

$$m_{\text{Total}} = m_{\text{Cloth}} + m_{\text{Hanger}} + m_{\text{Sling}} \tag{2.2}$$

F คือ แรงจากน้ำหนักรวมของโหลด (นิวตัน)
m_{Total} คือ น้ำหนักรวมของโหลด (กิโลกรัม)
m_{Cloth} คือ น้ำหนักของผ้า (กิโลกรัม)
m_{Hanger} คือ น้ำหนักของวงแขวนผ้า (กิโลกรัม)
m_{Sling} คือ น้ำหนักของเส้นเอ็น (กิโลกรัม)
g คือ ค่าคงที่โน้มถ่วงสากล (9.8 เมตรต่อวินาที)

2.17.2 การหาความเร็วของเฟืองที่มอเตอร์

จากทฤษฎีความเร็วเชิงมุม ในการหมุนแกนเพลาของมอเตอร์ที่ใช้ขับเคลื่อน รอกแขวนผ้า อัตราเร็วของมอเตอร์มีความสัมพันธ์กับค่าอัตราเร็วรอบ ดังสมการที่ (2.3) และ (2.4)

$$N_{\text{m}} = \frac{Vi}{2\pi r} \tag{2.3}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อจัดรูปสมการจะสามารถคำนวณหาความเร็วของเฟืองของมอเตอร์ที่ออกแบบได้ดังสมการที่ (3.4)

$$V = \frac{(2\pi r) \times N_{\mathrm{m}}}{i} \tag{2.4}$$

เมื่อ

i คือ อัตราทดรอบระหว่างเฟือง

r คือ รัศมีของเฟืองที่มอเตอร์ (เมตร)

V คือ ความเร็วของเฟืองที่มอเตอร์ (เมตรต่อวินาที)

N_{m} คือ อัตราเร็วรอบของมอเตอร์ (รอบต่อนาที)

2.17.3 การหากำลังของมอเตอร์

จากทฤษฎีงานและกำลังงาน กำลังเป็นตัวชี้วัดความสามารถในการทำงานของทั้งเครื่องยนต์ มนุษย์ สัตว์ หรือสิ่งมีชีวิตอื่น ๆ ส่วนงานเป็นปริมาณของพลังงานที่เป็นผลมาจากแรงซึ่งกระทำจากวัตถุ ก่อนส่งผลให้วัตถุดังกล่าวเคลื่อนที่ไปตามแนวแรงได้ในระยะทางหนึ่ง โดยงานและกำลังงานมีความสัมพันธ์กันดังสมการที่ (2.5)

$$P_{\mathrm{design}} = \frac{W}{t} \tag{2.5}$$

P_{design} คือ กำลังของมอเตอร์ที่ออกแบบ (วัตต์)

W คือ งานของมอเตอร์ (จูล)

t คือ เวลาที่มอเตอร์ใช้ (วินาที)

สามารถเขียนงานอยู่ในรูปแบบของความสัมพันธ์ของแรงและระยะทางได้ดังสมการที่ (2.6)

$$P_{\mathrm{design}} = \frac{F \times S}{t} \tag{2.6}$$

F คือ แรงจากน้ำหนักรวมของโหลด (นิวตัน)

S คือ ระยะทางที่เคลื่อนที่ได้จากแรง (เมตร)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดังนั้น สามารถพิจารณาค่ากำลังงานในขณะที่วัตถุเคลื่อนที่ด้วยความเร็วค่าหนึ่ง ได้ดังสมการที่ (2.7)

$$P_{\text{design}} = F \times V \tag{2.7}$$

V คือ ความเร็วของเฟืองที่มอเตอร์ (เมตรต่อวินาที)

2.17.4 การหาแรงบิดของมอเตอร์

สำหรับการออกแบบมอเตอร์ ควบคุมเลื่อนเข้าเลื่อนออกของรอก จำเป็นต้องคำนึงถึงรอกแขวน น้ำหนักของผ้า ไม้แขวนผ้า เส้นเอ็นที่ใช้เคลื่อนที่ และอัตราเร็วของมอเตอร์ โดยนำตัวแปรเหล่านี้มาคำนวณหาค่าของอัตราเร็วรอบและแรงบิดของมอเตอร์ สามารถหาได้ดังสมการที่ (2.8) เพื่อใช้ในการพิจารณาเลือกมอเตอร์ที่เหมาะสมกับชิ้นงาน

$$P = \frac{2\pi N_{\text{m}} T}{60} \tag{2.8}$$

P คือ กำลังของมอเตอร์ (วัตต์)
N_{m} คือ อัตราเร็วรอบของมอเตอร์ (รอบต่อนาที)
T คือ แรงบิดของมอเตอร์ (นิวตันเมตร)

2.18 ระบบปฏิบัติการ Android

ระบบปฏิบัติการแบบเปิดเผยซอฟต์แวร์ต้นฉบับ (Open Source) โดยบริษัท Google (Google Inc.) ที่ได้รับความนิยมเป็นอย่างสูง เนื่องจากอุปกรณ์ที่ใช้ระบบปฏิบัติการ Android มีจำนวนมาก อุปกรณ์มีหลากหลายระดับ หลายราคา รวมทั้งสามารถทำงานบนอุปกรณ์ที่มีขนาดหน้าจอ และความละเอียดแตกต่างกันได้ ทำให้ผู้ใช้งานสามารถเลือกได้ตามต้องการ

สำหรับนักพัฒนาโปรแกรม (Programmer) แล้วนั้น การพัฒนาโปรแกรมเพื่อใช้งานบนระบบปฏิบัติการ Android ไม่ใช่เรื่องที่ยากเพราะมีข้อมูลในการพัฒนารวมทั้ง Android SDK (Software Development Kit) เตรียมไว้ให้กับนักพัฒนาได้เรียนรู้และเมื่อนักพัฒนาต้องการจะเผยแพร่หรือจำหน่ายโปรแกรมที่พัฒนาแล้วเสร็จ Android ก็ยังมีตลาดในการเผยแพร่โปรแกรมผ่าน Android Market แต่หากจะกล่าวถึงโครงสร้างภาษาที่ใช้ในการพัฒนานั้น สำหรับ Android SDK จะยึดโครงสร้างของภาษา Java (Java language) ในการเขียนโปรแกรม [20]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.19 Android Studio

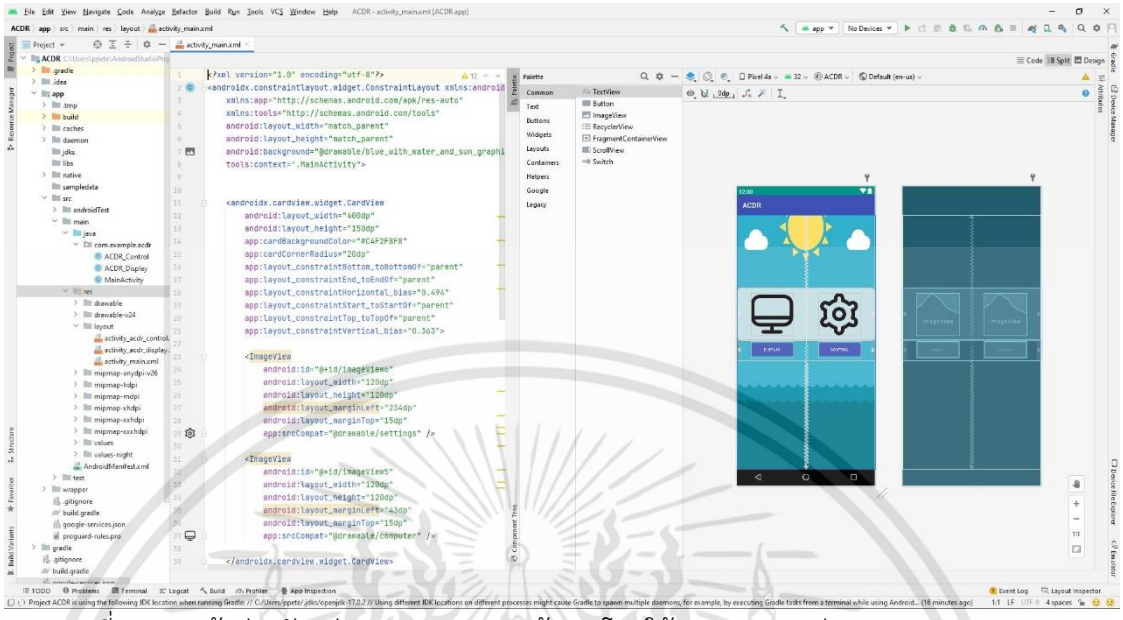
Android Studio เป็น IDE Tool จาก Google ไว้พัฒนา Android สำหรับ Android Studio เป็น IDE Tools ล่าสุดจาก Google ไว้พัฒนาโปรแกรม Android โดยเฉพาะ โดยพัฒนาจากแนวคิดพื้นฐานมาจาก InteliJ IDEA คล้ายกับการทำงานของ Eclipse และ Android ADT Plugin โดยวัตถุประสงค์ของ Android Studio คือต้องการพัฒนาเครื่องมือ IDE ที่สามารถพัฒนาแอปพลิเคชันได้บน Android ให้มีประสิทธิภาพมากขึ้น ทั้งด้านการออกแบบ GUI ที่ช่วยให้สามารถพรีวิวตัวแอปพลิเคชันได้ มุมมองที่แตกต่างกันบน Smart Phone แต่ล่ะรุ่น สามารถแสดงผลบางอย่างได้ทันที โดยไม่ต้องทำการออกคำสั่งเพื่อใช้งานแอปพลิเคชันได้บน Emulator รวมทั้งยังแก้ไขปรับปรุงในเรื่องของความเร็วของ Emulator ที่ยังเจอปัญหากันอยู่ในปัจจุบัน [21]

2.19.1 ส่วนต่อประสานผู้ใช้ (User Interface)

ส่วนต่อประสานผู้ใช้ (User Interface) คือ ส่วนที่ผู้ใช้สามารถมองเห็นและโต้ตอบได้ โดย Android Studio มีส่วนประกอบของส่วนต่อประสานผู้ใช้สำเร็จรูปมากมาย เช่น แบบแผนที่มีโครงสร้างและการควบคุมส่วนต่อประสานงานผู้ใช้ที่ช่วยให้สามารถสร้างและออกแบบกราฟิกสำหรับส่วนต่อประสานผู้ใช้ภายในแอปพลิเคชันได้ อีกทั้ง Android Studio ยังมีโมดูลส่วนต่อประสานผู้ใช้อื่น ๆ สำหรับออกแบบส่วนต่อประสานพิเศษ เช่น กล่องโต้ตอบ การแจ้งเตือน และเมนูโดยจะใช้ภาษา XML ในการสร้างโครงสร้าง [22]

XML หรือ Extensible Markup Language เป็นภาษาที่ใช้สำหรับแสดงผลข้อมูล โดยแตกต่างจากภาษา HTML โดยภาษา HTML ถูกออกแบบมาสำหรับการแสดงผลเท่านั้น ส่วนภาษา XML ถูกออกแบบมาเพื่อโครงสร้างและข้อมูล ซึ่งภาษาที่ใช้จะกำหนดรูปแบบของคำสั่งภาษา HTML หรือเรียกว่า Meta Data ใช้สำหรับกำหนดรูปแบบคำสั่ง Markup ต่าง ๆ โดยภาษา XML สามารถจัดการด้านระบบการติดต่อกับผู้ใช้จากโครงสร้างของข้อมูลเองได้และอาจจะต้องใช้ร่วมกับภาษาอื่น เช่น HTML JSP PHP ASP หรือภาษาอื่นๆ ที่สนับสนุน XML เพื่อให้สามารถแสดงผลได้อย่างถูกต้อง โดยจะมีนามสกุลเป็น .XML ซึ่งหน้าจอแอปพลิเคชันสามารถออกแบบร่วมกันได้ทั้งจาก XML และ Virtual แบบลากวางได้ ทำให้สะดวกในการออกแบบและใช้งาน [23] หน้าต่างการออกแบบแสดงได้ดังรูปที่ 2.17

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

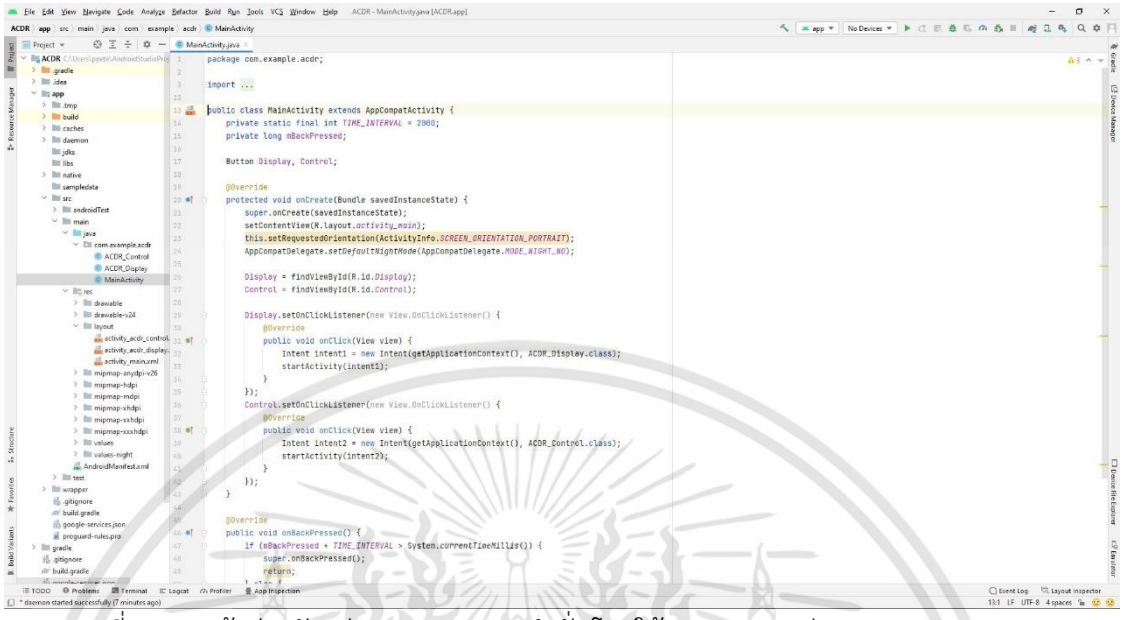


รูปที่ 2.17 หน้าต่างตัวอย่างการออกแบบหน้าจอ โดยใช้ภาษา XML ผ่าน Android Studio

2.20 Java

Java หรือ Java programming Language เป็นภาษาสำหรับเขียนโปรแกรมที่สนับสนุนการเขียนโปรแกรมเชิงวัตถุ (Object-Oriented Programming: OOP) โดยมีจุดประสงค์เพื่อใช้แทนภาษา C++ ที่มีความซับซ้อนมากกว่าและมีจำนวนคำสั่งมากกว่าทำให้เกิดข้อผิดพลาดได้ง่าย ซึ่งรูปแบบที่เพิ่มขึ้นมีความคล้ายคลึงกับภาษา Objective-C โดยจุดเด่นของภาษา Java นักพัฒนาสามารถใช้หลักของ Object-Oriented Programming มาพัฒนาโปรแกรมของตัวเองด้วยภาษา Java ได้ เหมาะสำหรับพัฒนาระบบที่มีความซับซ้อน สามารถทำงานในระบบปฏิบัติการที่แตกต่างกันได้ อีกทั้งภาษา Java ยังมีการตรวจสอบข้อผิดพลาดในขณะที่ compile time และ runtime สามารถช่วยลดข้อผิดพลาดที่อาจเกิดขึ้นในการเขียนโปรแกรมและทำให้ debug โปรแกรมได้ง่าย หน้าต่างตัวอย่างการออกแบบคำสั่งภาษา Java ผ่าน Android Studio แสดงดังรูปที่ 2.18 [24]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.18 หน้าต่างตัวอย่างการออกแบบคำสั่ง โดยใช้ภาษา Java ผ่าน Android Studio

2.21 Firebase

Firebase คือแพลตฟอร์มที่รวบรวมเครื่องมือต่าง ๆ สำหรับการจัดการในส่วนของ Backend หรือ Server side ซึ่งทำให้สามารถสร้างแอปพลิเคชันสำหรับอุปกรณ์สื่อสารได้อย่างมีประสิทธิภาพ ช่วยลดเวลาและค่าใช้จ่ายในการทำ Server side หรือการวิเคราะห์ข้อมูล Firebase เป็นฐานข้อมูลแบบ NoSQL คือไม่ใช้ภาษา SQL ในการจัดการข้อมูล แต่จะเน้นไปที่ความเร็วในการใช้งาน มีการเก็บข้อมูลแบบ JSON โดยที่มีตารางเหมือนกับ SQL แต่ไม่มีหลัก ในแถวสามารถเก็บข้อมูลได้ทั้งข้อความ (String) ตัวเลข (Number) และอื่น ๆ ได้ Firebase ยังมีช่องทางการเชื่อมต่อที่สามารถรองรับหลายภาษาเช่น Python บน Raspberry Pi JavaScript สำหรับใช้งานบนหน้าเว็บไซต์ ESP8266 และ NodeMCU ESP8266 อีกด้วย [25]

2.21.1 Realtime Database

Realtime Database คือบริการฐานข้อมูล NoSQL Cloud Database ที่ใช้วิธีการเก็บข้อมูลในรูปแบบ JSON และสามารถเชื่อมต่อข้อมูลแบบ Realtime Database กับทุกอุปกรณ์ และสามารถออกแบบเงื่อนไขการเข้าถึงข้อมูลทั้งการอ่านและเขียนได้ทั้งในระบบปฏิบัติการ Android IOS และ Web Application [26]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

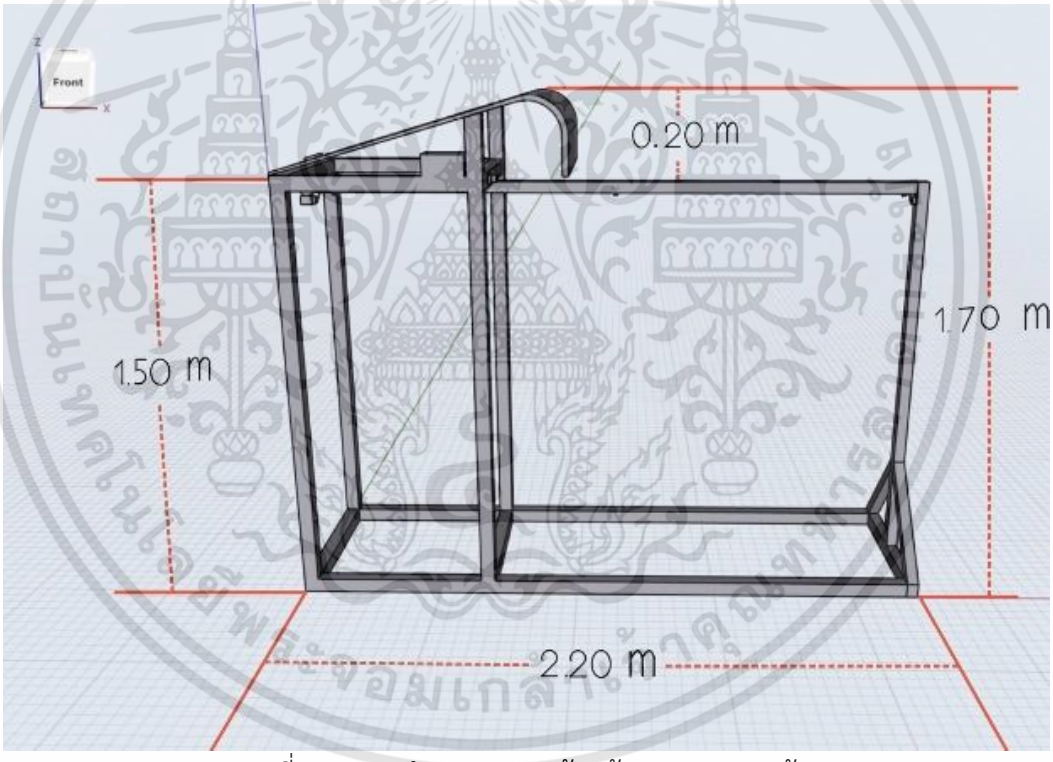
บทที่ 3

การออกแบบและการจัดทำปริญญานิพนธ์

3.1 การออกแบบ

3.1.1 การออกแบบโครงสร้างของราวตากผ้า

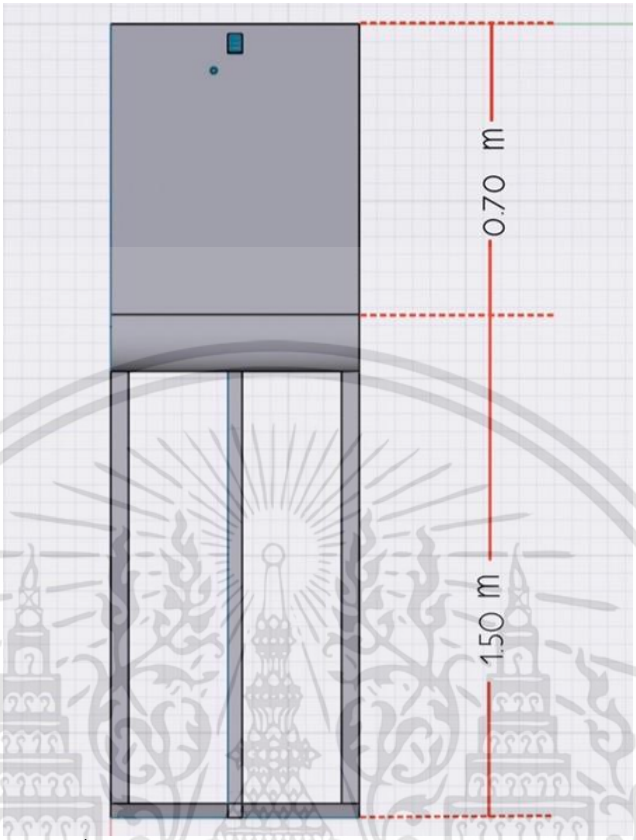
ในการออกแบบโครงสร้างของราวตากผ้าในส่วนของหลังคาจะออกแบบเป็นสามเหลี่ยมมุมฉาก เพื่อที่จะให้น้ำฝนไหลตามพื้นเอียง และใช้ราวอะลูมิเนียมสำหรับตากผ้าทั้งหมด 1 ราว การออกแบบโครงสร้างของราวตากผ้าในโปรแกรม Shapr3D ขนาดของราวตากผ้าที่ออกแบบไว้มีความกว้าง 0.70 เมตร ความยาว 2.20 เมตร และความสูง 1.70 เมตร ดังรูปที่ 3.1



รูปที่ 3.1 ภาพจำลองมุมมองด้านข้างของราวตากผ้า

รูปที่ 3.2 แสดงมุมมองด้านบนของราวตากผ้า โดยตู้เก็บผ้ามีความยาว 0.70 เมตร และส่วนที่นำผ้าออกมาตากแดดมีความยาว 1.50 เมตร

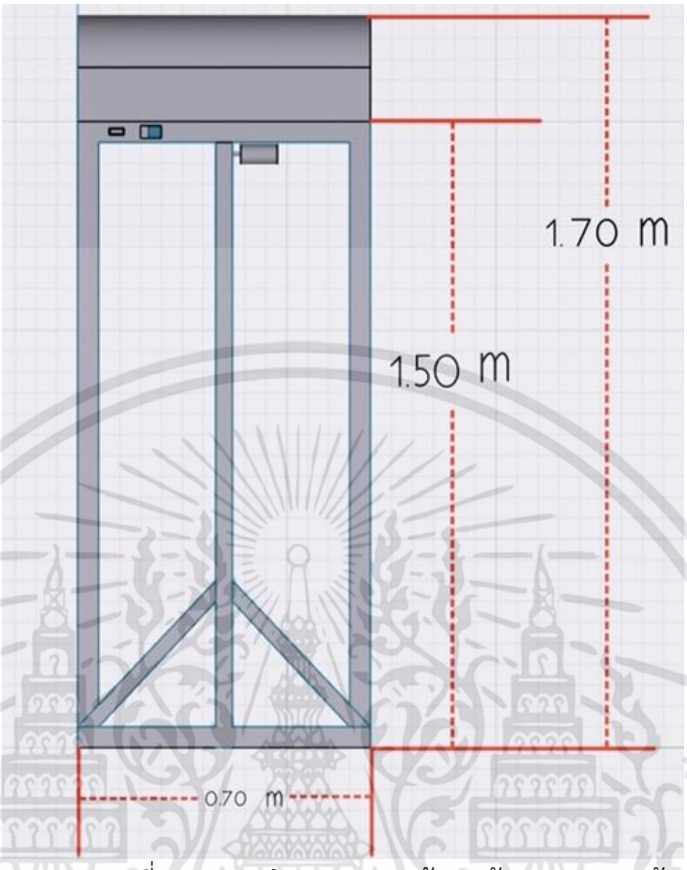
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.2 ภาพจำลองมุมมองด้านบนของราวตากผ้า

รูปที่ 3.3 แสดงมุมมองด้านหน้าของราวตากผ้า สังเกตได้ว่าช่วงฐานที่มีโครงสร้างเป็นรูปสามเหลี่ยมออกแบบมาเพื่อช่วยรองรับน้ำหนักในส่วนของแท่งเหล็กตรงกลางโดยเชื่อมต่อกับอะลูมิเนียมในแนวตั้ง มีทิศทางตั้งขึ้นในแนวตั้งเพื่อเชื่อมต่อกับคานอะลูมิเนียม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

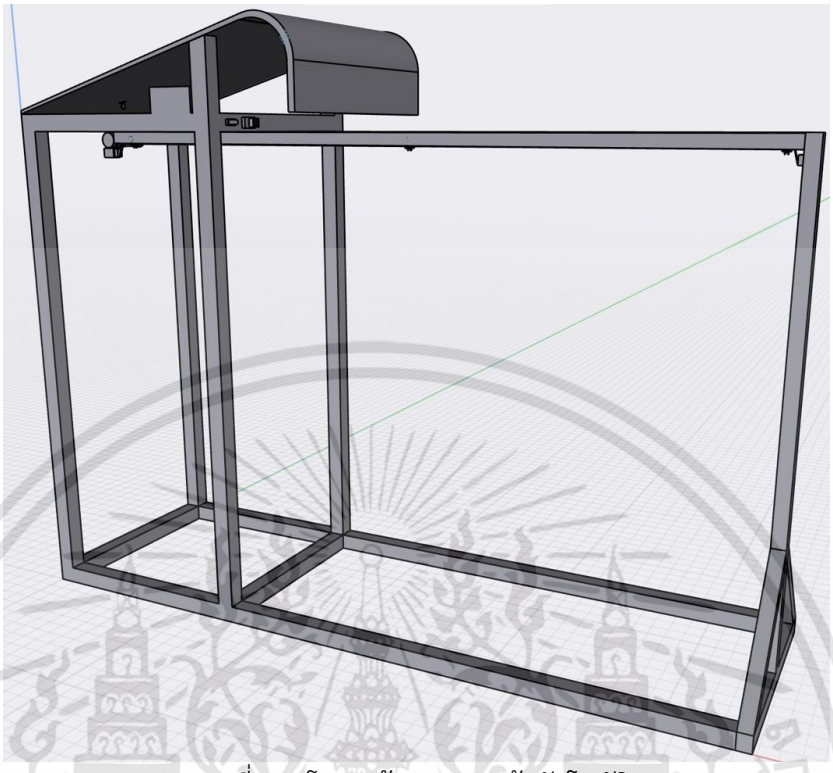


รูปที่ 3.3 ภาพจำลองมุมมองด้านหน้าของราวตากผ้า

3.1.2 ออกแบบการวางตำแหน่งของอุปกรณ์

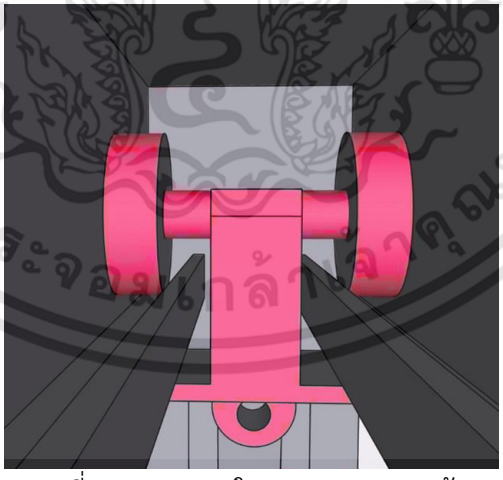
ราวตากผ้าอัตโนมัติออกแบบผ่านทางแอปพลิเคชัน Shapr 3D CAD modeling ซึ่งโปรแกรมสามารถจำลองการวางตำแหน่งของอุปกรณ์ต่าง ๆ ประกอบด้วย รอก กล่องใส่อุปกรณ์ มอเตอร์ ลิมิตสวิตช์ โมดูลตรวจจับน้ำฝน เซนเซอร์วัดความชื้นและอุณหภูมิ และเซนเซอร์ตรวจจับความเข้มแสง แสดงดังรูปที่ 3.4 ถึง 3.10

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.4 โครงสร้างราวตากผ้าอัตโนมัติ

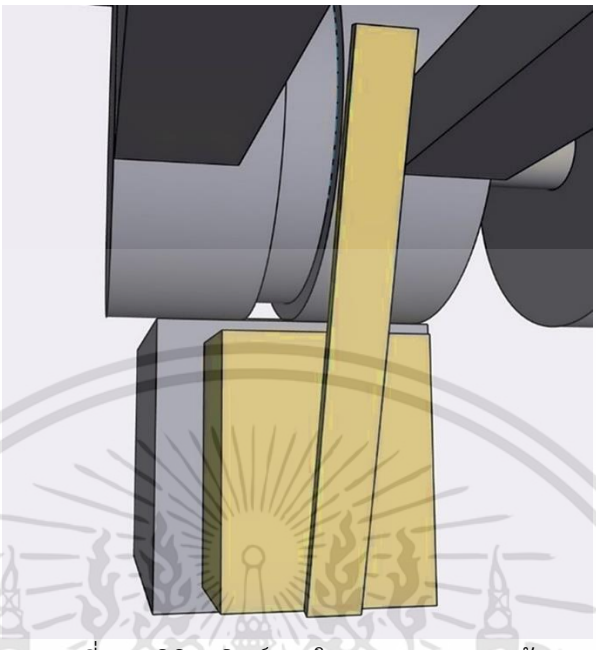
1.ลักษณะของรอก อยู่ภายในรางของราวตากผ้า แสดงดังรูปที่ 3.5



รูปที่ 3.5 รอกภายในรางของราวตากผ้า

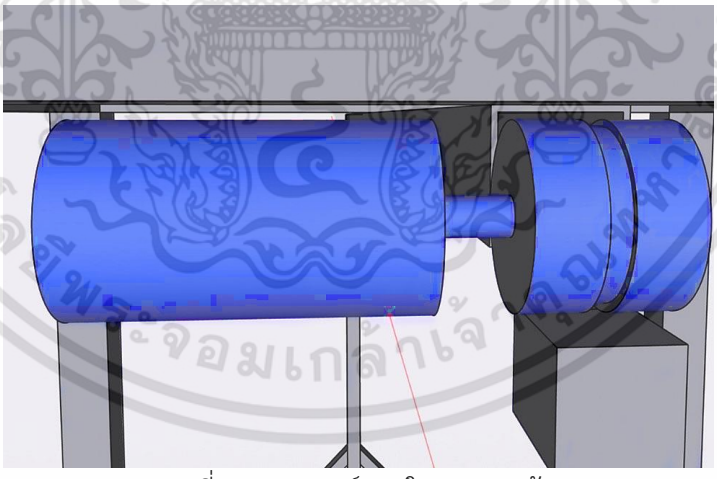
2.ลิมิตสวิตช์ ให้ตัวก้านของลิมิตสวิตช์อยู่ภายในราง เพื่อตัดการทำงานเมื่อรอกเลื่อนมาสัมผัสลิมิตสวิตช์ แสดงดังรูปที่ 3.6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.6 ลิมิตสวิตช์ภายในรางของราวตากผ้า

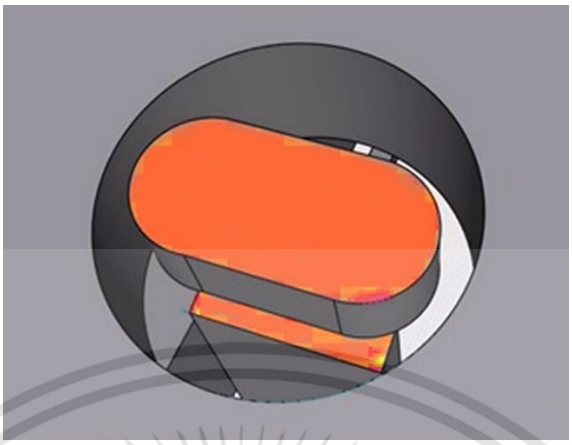
3.การติดตั้งมอเตอร์ ให้เพลามอเตอร์อยู่พอดีกับรางของราวตากผ้า เพื่อชักรอกเข้า และออก แสดงดังรูปที่ 3.7



รูปที่ 3.7 มอเตอร์ภายในราวตากผ้า

4.เซนเซอร์ตรวจวัดความเข้มแสง ติดตั้งจุดตรวจสอบค่าความเข้มแสงบริเวณด้านบนของหลังคา แสดงดังรูปที่ 3.8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



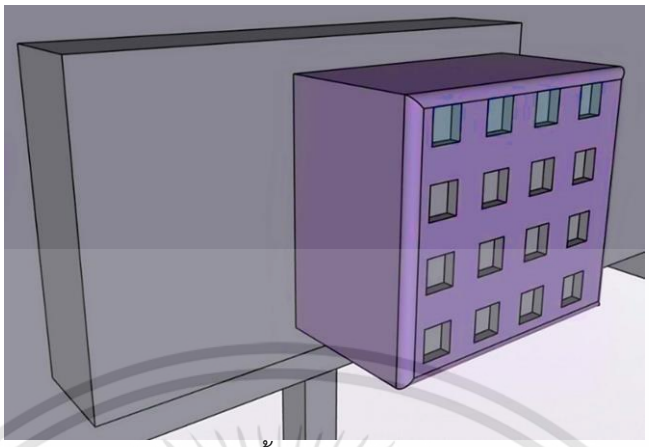
รูปที่ 3.8 เซนเซอร์ตรวจวัดความเข้มแสงบริเวณด้านบนของหลังคา

5.โมดูลตรวจจับน้ำฝน ติดตั้งจุดตรวจจับน้ำฝนบริเวณด้านบนของหลังคา แสดงดังรูปที่ 3.9

รูปที่ 3.9 โมดูลตรวจจับน้ำฝนบริเวณด้านบนของหลังคา

6.เซนเซอร์วัดความชื้นและอุณหภูมิ ติดตั้งจุดตรวจสอบบริเวณชายคาของหลังคา แสดงดังรูปที่ 3.10

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

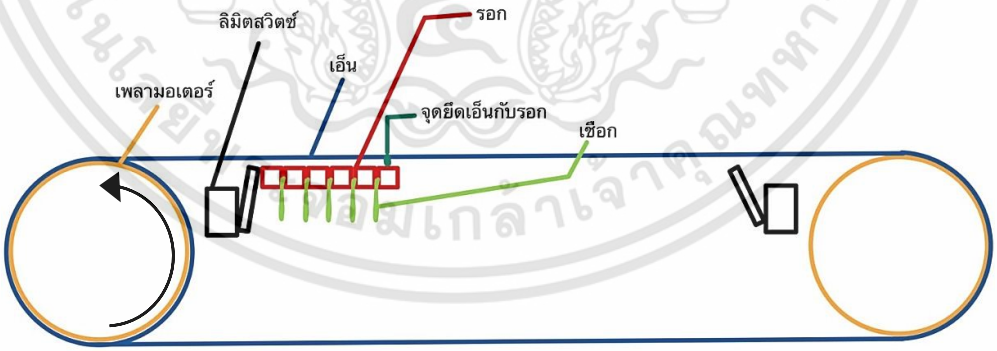


รูปที่ 3.10 เซนเซอร์วัดความชื้นและอุณหภูมิบริเวณชายคาของหลังคา

3.1.3 ออกแบบการหมุนรอก

การออกแบบรอกแขวนผ้าระยะห่างของผ้าขณะตากจะมีระยะห่าง 10 เซนติเมตร เท่า ๆ กัน และเมื่อเครื่องทำการเก็บผ้าเข้าสู่ตู้จะทำให้ผ้าทุกตัวติดกันโดยใช้เชือกผูกกับรอกเข้าด้วยกัน และยึดรอกตัวสุดท้ายติดกับเอ็นไว้ ทิศทางการเคลื่อนที่ของรอกจะแสดงด้วยลูกศรสีดำ แสดงดังรูปที่ 3.11 และรูปที่ 3.12

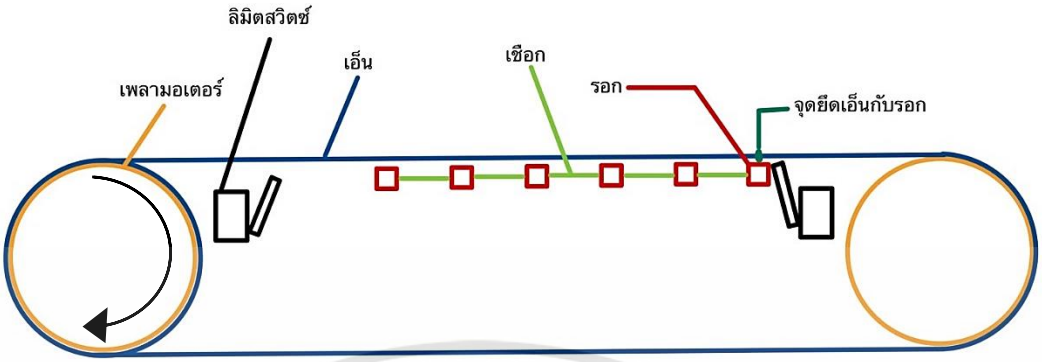
การเลื่อนรอกเข้า เพลามอเตอร์หมุนทวนเข็มนาฬิกาจนกระทั่งรอกตัวแรกสัมผัสลิมิตสวิตช์ โดยรอกตัวสุดท้ายที่ยึดกับเอ็นเป็นตัวดันรอกตัวอื่น ๆ ให้เข้าไปในที่เก็บผ้า และเมื่อรอกตัวแรกสัมผัสลิมิตสวิตช์ มอเตอร์จะหยุดหมุน



รูปที่ 3.11 การเลื่อนรอกเข้า

การเลื่อนรอกออก เพลามอเตอร์หมุนตามเข็มนาฬิกาจนกระทั่งรอกตัวแรกสัมผัสลิมิตสวิตช์ โดยรอกตัวสุดท้ายที่ยึดกับเอ็นเป็นตัวดึงรอกตัวอื่น ๆ ด้วยเชือกให้ออกไปนอกที่เก็บผ้า และเมื่อรอกตัวสุดท้ายสัมผัสลิมิตสวิตช์ มอเตอร์จะหยุดหมุน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.12 การเลื่อนรอกออก

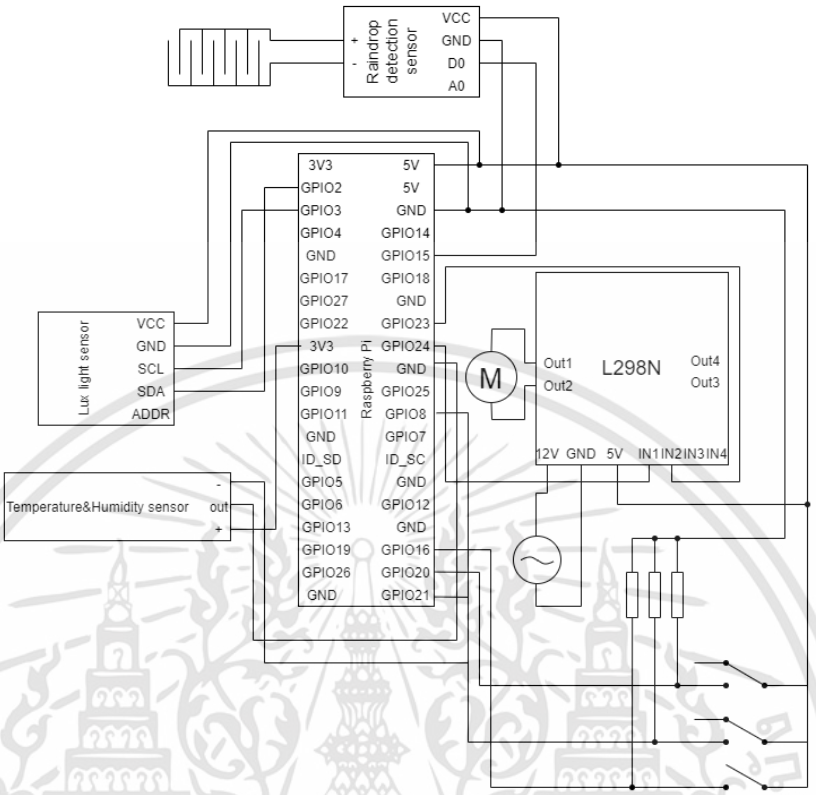
3.1.4 ออกแบบวงจรการเชื่อมต่ออุปกรณ์

การออกแบบวงจรการเชื่อมต่อใช้บอร์ดไข่ปลาในการเชื่อมต่ออุปกรณ์ทั้งหมดเนื่องจากพอร์ตแหล่งจ่ายแรงดันไฟฟ้า 5 โวลต์ และ GND ของ Raspberry Pi มีไม่เพียงพอกับอุปกรณ์รวมถึงตัวต้านทานที่ใช้งานไม่สามารถเชื่อมต่อได้ จึงนำตัวต้านทาน แหล่งจ่ายแรงดันไฟฟ้า 5 โวลต์ และ GND เชื่อมกับบอร์ดไข่ปลา

การเชื่อมอุปกรณ์ทั้งหมดเข้าด้วยกันประกอบด้วย เซนเซอร์ตรวจวัดความเข้มแสง โมดูลตรวจจับน้ำฝน เซนเซอร์วัดความชื้นและอุณหภูมิ โมดูลขับมอเตอร์ และลิมิตสวิตช์ เข้าด้วยกันแสดงดังรูปที่ 3.13 อุปกรณ์ทั้งหมดต่อเข้ากับแรงดันไฟฟ้าขั้วบวกขนาด 5 โวลต์ ส่วนแรงดันไฟฟ้า 12 โวลต์ จากแบตเตอรี่ต่อเข้าโมดูลขับมอเตอร์ โดยเซนเซอร์ตรวจวัดความเข้มแสงกำหนดใช้ขา GPIO 2 และ GPIO 3 โมดูลตรวจจับน้ำฝนกำหนดใช้ขา GPIO 15 เซ็นเซอร์วัดความชื้นและอุณหภูมิกำหนดใช้ขา GPIO 8

โมดูลขับมอเตอร์กำหนด GPIO 23 เพื่อควบคุมการหมุนในทิศทางตามเข็มนาฬิกาและ GPIO 24 เพื่อควบคุมการหมุนในทิศทางทวนเข็มนาฬิกา ลิมิตสวิตช์กำหนดใช้ขา GPIO 20 และ GPIO 21 เพื่อควบคุมการเปิดปิดกำหนดใช้ขา GPIO 16

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



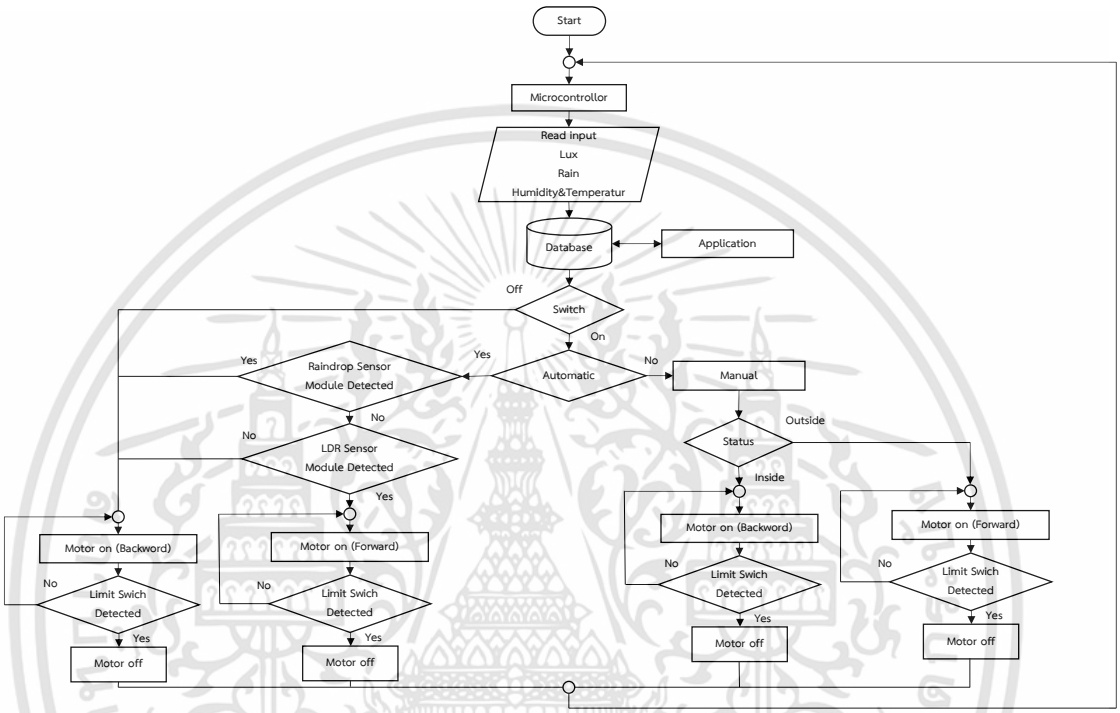
รูปที่ 3.13 การเชื่อมต่ออุปกรณ์ต่าง ๆ

3.1.5 ออกแบบชุดคำสั่ง

ออกแบบชุดคำสั่งการทำงานของอุปกรณ์ทั้งหมดประกอบด้วยเซนเซอร์ตรวจวัดความเข้มแสง โมดูลตรวจจับน้ำฝน เซนเซอร์วัดความชื้นและอุณหภูมิ โมดูลขับมอเตอร์ และลิมิตสวิตช์ทำการเชื่อมต่ออุปกรณ์ทั้งหมดเข้าด้วยกัน โดยชุดคำสั่งที่ระบุเงื่อนไขการทำงานของมอเตอร์ตามสภาพภูมิอากาศที่เซนเซอร์ตรวจวัดได้ คือ เมื่อ Raspberry Pi ทำงาน ถ้าสวิตช์อยู่ในสถานะปิดมอเตอร์จะหมุนเลื่อนลูกล้อรางม่านเคลื่อนที่เข้าด้านในจนสัมผัสกับลิมิตสวิตช์ด้านใน แต่ถ้าสวิตช์อยู่ในสถานะเปิด จะทำการตรวจสอบโหมดการทำงานว่าเป็นการทำงานอัตโนมัติ หรือการใช้งานระบบสั่งการด้วยมือ ถ้าเป็นการทำงานอัตโนมัติ เซนเซอร์ตรวจจับน้ำฝนจะทำงาน ถ้าเซนเซอร์ตรวจจับได้ว่ามีน้ำ มอเตอร์จะหมุนเลื่อนลูกล้อรางม่านเคลื่อนที่เข้าด้านในจนสัมผัสกับลิมิตสวิตช์ด้านใน แต่ถ้าเซนเซอร์ตรวจจับได้ว่าไม่มีน้ำ จะพิจารณาเงื่อนไขของเซนเซอร์ตรวจจับความเข้มแสงต่อไป ถ้าเซนเซอร์ตรวจวัดได้ว่าความเข้มแสงมีค่ามากกว่า 0 ลักซ์ มอเตอร์จะหมุนเลื่อนลูกล้อรางม่านเคลื่อนที่ออกจนสัมผัสกับลิมิตสวิตช์ด้านนอก แต่ถ้าเซนเซอร์ตรวจวัดได้ว่าความเข้มแสงมีค่าเป็น 0 ลักซ์ มอเตอร์จะหมุนเลื่อนลูกล้อรางม่านเคลื่อนที่เข้าด้านในจนสัมผัสกับลิมิตสวิตช์ด้านใน แต่ถ้าเป็นการใช้งานระบบสั่งการด้วยมือ จะเป็นการสั่งการผ่านแอปพลิเคชัน ถ้ามีสถานะเป็น Inside มอเตอร์จะหมุน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เลื่อนลูกล้อรางม่านเคลื่อนที่เข้าด้านในจนสัมผัสกับลิมิตสวิตช์ด้านใน แต่ถ้ามีสถานะเป็น Outside มอเตอร์จะหมุนเลื่อนลูกล้อรางม่านเคลื่อนที่ออกจนสัมผัสกับลิมิตสวิตช์ด้านนอก ทำงานเช่นนี้วนรอบไปตลอดเวลา สามารถเขียนได้เป็นแผนผังหลักการทำงานของราวตากผ้าอัตโนมัติแสดงดังรูปที่ 3.14



รูปที่ 3.14 แผนผังหลักการทำงานของราวตากผ้าอัตโนมัติ

3.1.6 การออกแบบมอเตอร์

3.1.5.1 การออกแบบแรงจากน้ำหนักรวมของโหลด

ในการออกแบบน้ำหนักรวมของโหลด ผู้จัดทำได้ทำการกำหนดค่าคงที่โน้มถ่วงสากลเป็น 9.81 เมตรต่อวินาทีกำลังสอง น้ำหนักของผ้าเป็น 10 กิโลกรัม น้ำหนักของวงแขวนผ้าเป็น 0.25 กิโลกรัม และน้ำหนักของเส้นเอ็นเป็น 0.25 กิโลกรัม ให้สมการของแรงจากน้ำหนักรวมของโหลดแสดงดังสมการที่ (3.1) และ (3.2)

$$F = m_{\text{Total}} \times g \tag{3.1}$$

สามารถหาค่า m_{Total} ได้จากสมการที่ (3.2)

$$m_{\text{Total}} = m_{\text{Cloth}} + m_{\text{Hanger}} + m_{\text{Sling}} \tag{3.2}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แทนค่าตัวแปรในสมการที่ (3.1) และ (3.2) โดยที่ F คือ แรงจากน้ำหนักรวมของโหลด m_{Total} คือ น้ำหนักรวมของโหลด m_{Cloth} คือ น้ำหนักของผ้า m_{Hanger} คือ น้ำหนักของวงแขวนผ้า m_{Sling} คือ น้ำหนักของเส้นเอ็น g คือ ค่าคงที่โน้มถ่วงสากล ได้ผลลัพธ์ดังนี้

$$m_{\text{Total}} = 10 + 0.25 + 0.25$$
$$m_{\text{Total}} = 10.50 \text{ กิโลกรัม}$$
$$F = 10.50 \times 9.81$$
$$F = 103.01 \text{ นิวตัน}$$

3.1.5.2 การออกแบบความเร็วของเฟืองที่มอเตอร์

ในการออกแบบความเร็วของเฟืองที่มอเตอร์ ผู้จัดทำได้ทำการกำหนดค่า π เป็น 3.14 รัศมีของเพลามอเตอร์เป็น 0.03 เมตร อัตราทดรอบระหว่างเฟืองเป็น 60 และความเร็วรอบของมอเตอร์เป็น 80 รอบต่อนาที ให้สมการความเร็วของเฟืองที่มอเตอร์แสดงดังสมการที่ (3.3)

$$V = \frac{(2\pi r) \times N_{\text{m}}}{i} \tag{3.3}$$

แทนค่าตัวแปรในสมการที่ (3.4) โดยที่ V คือ ความเร็วของเฟืองที่มอเตอร์ r คือ รัศมีของเฟืองที่มอเตอร์ i คือ อัตราทดรอบระหว่างเฟือง และ N_{m} คือ อัตราเร็วรอบของมอเตอร์ ได้ผลลัพธ์ดังนี้

$$V = \frac{(2 \times 3.14 \times 0.03) \times 80}{60}$$
$$V = 0.25 \text{ เมตรต่อวินาที}$$

3.1.5.3 การออกแบบกำลังของมอเตอร์

ในการออกแบบกำลังของมอเตอร์ ผู้จัดทำได้กำหนดค่าความเร็วเพลามอเตอร์ที่ได้จากสมการที่ (3.3) เป็น 0.25 เมตรต่อวินาที และแรงจากน้ำหนักรวมของโหลดที่ได้จากสมการที่ (3.1) เป็น 103.01 นิวตัน ให้สมการกำลังของมอเตอร์แสดงดังสมการที่ (3.4)

$$P_{\text{design}} = F \times V \tag{3.4}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แทนค่าตัวแปรในสมการที่ (3.4) โดยที่ P_{design} คือ กำลังของมอเตอร์ที่ออกแบบ F คือ แรงจากน้ำหนักรวมของโหลด และ V คือ ความเร็วของเฟืองที่มอเตอร์ ได้ผลลัพธ์ดังนี้

$$P_{\text{design}} = 103.01 \times 0.25$$
$$P_{\text{design}} = 25.75 \text{ วัตต์}$$

3.1.7 การออกแบบชุดคำสั่งการรับส่งข้อมูลไปยังฐานข้อมูล

ออกแบบชุดคำสั่งการรับส่งข้อมูลและการเชื่อมต่อกับฐานข้อมูล Firebase Realtime Database โดยสร้างโครงงานทดลองที่มีชื่อว่า testfire เพื่อทำการรับข้อมูลจาก Raspberry Pi และเชื่อมต่อกับแอปพลิเคชันบน Android โดยได้สร้างที่อยู่ของไฟล์ไว้สำหรับเชื่อมต่อข้อมูลจากเซนเซอร์ต่าง ๆ ไปแสดงบนแอปพลิเคชัน และสำหรับการควบคุมมอเตอร์ ซึ่งจะมีชื่อที่อยู่ของไฟล์แทนคำสั่งทั้งหมดดังต่อไปนี้ แสดงดังรูปที่ 3.15

1. humidity หมายถึง การแสดงค่าความชื้นในอากาศในหน่วยเปอร์เซ็นต์จากเซนเซอร์วัดความชื้นและอุณหภูมิให้แสดงบนหน้าแสดงผลการทำงานในแอปพลิเคชัน ณ ขณะนั้น

2. intensity หมายถึง การแสดงค่าความเข้มแสงในหน่วยลักซ์จากเซนเซอร์วัดความเข้มแสงให้แสดงบนหน้าแสดงผลการทำงานในแอปพลิเคชัน ณ ขณะนั้น

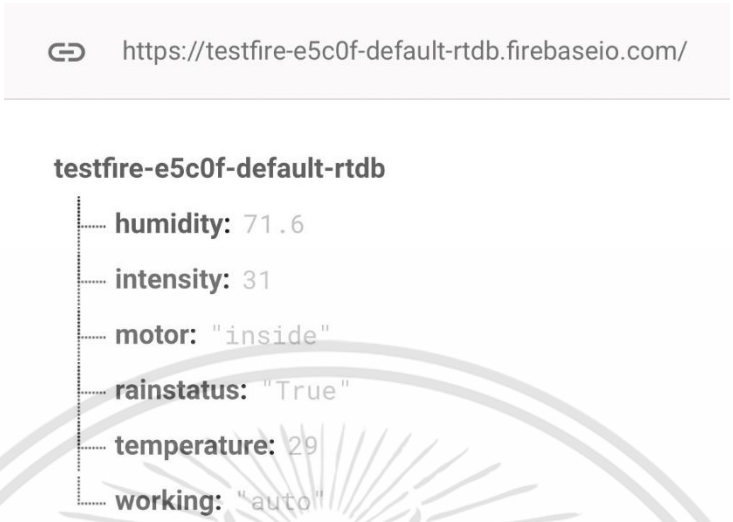
3. motor หมายถึง การกำหนดทิศทางการหมุนขอมอเตอร์ ถ้ามีสถานะเป็น inside หมายถึง ให้มอเตอร์หมุนเคลื่อนที่เข้าด้านในจนสัมผัสกับลิมิตสวิตช์ด้านใน และถ้ามีสถานะเป็น outside หมายถึง ให้มอเตอร์หมุนเคลื่อนที่ออกด้านนอกจนสัมผัสกับลิมิตสวิตช์ด้านนอก

4. rainstatus หมายถึง การแสดงสถานะการตรวจจับน้ำฝนจากโมดูลตรวจจับน้ำฝนให้แสดงบนหน้าแสดงผลการทำงานในแอปพลิเคชัน ณ ขณะนั้น

5. temperature หมายถึง การแสดงค่าอุณหภูมิในอากาศในหน่วยองศาเซลเซียสจากเซนเซอร์วัดความชื้นและอุณหภูมิให้แสดงบนหน้าแสดงผลการทำงานในแอปพลิเคชัน ณ ขณะนั้น

6. working หมายถึง การแสดงโหมดการทำงานของราวตากผ้า ถ้ามีสถานะเป็น auto หมายถึง ให้ราวตากผ้าทำงานในโหมดอัตโนมัติ และถ้ามีสถานะเป็น manual หมายถึง ให้ราวตากผ้าทำงานในโหมดการสั่งการด้วยมือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.15 ชื่อที่อยู่ของไฟล์ที่เชื่อมต่อระหว่างอุปกรณ์และ Firebase

3.1.8 การออกแบบแอปพลิเคชัน

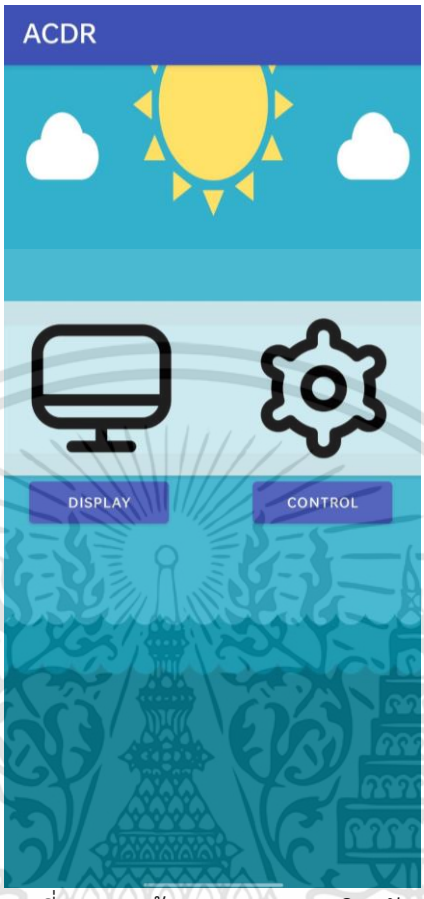
3.1.7.1 การออกแบบส่วนต่อประสานผู้ใช้งาน

จากการศึกษาได้มีการทดลองออกแบบแอปพลิเคชันสำหรับควบคุมราวตากผ้าอัตโนมัติ โดยได้ทำการออกแบบหน้าใช้งานสำหรับผู้ใช้งานดังนี้

1. หน้าแรกของแอปพลิเคชัน

เมื่อเข้าสู่แอปพลิเคชันจะพบกับหน้าแรกของแอปพลิเคชัน โดยผู้ใช้งานสามารถเลือกดูประเภทของการทำงานทั้งหมดได้ 2 โหมด คือ DISPLAY และ CONTROL แสดงดังรูปที่ 3.16

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

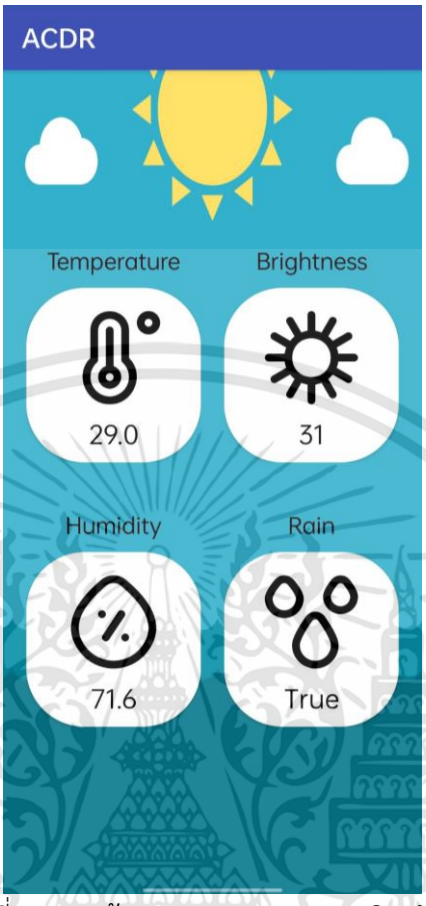


รูปที่ 3.16 หน้าแรกของแอปพลิเคชัน

2. หน้า DISPLAY ของแอปพลิเคชัน

ทำการออกแบบส่วนต่อประสานผู้ใช้งานสำหรับแสดงผลการทำงานของระบบควบคุมโดยกำหนดให้แสดงค่าเซนเซอร์ต่าง ๆ ที่ได้รับ ได้แก่ ค่าอุณหภูมิ ค่าความสว่าง ค่าความชื้นสัมพัทธ์ และค่าสถานะของฝนโดยทุกค่าจะถูกแสดงตามเวลาจริงทั้งหมด เพื่อให้ผู้ใช้งานได้รับรู้ข้อมูลอย่างรวดเร็วและสะดวกต่อการควบคุม แสดงดังรูปที่ 3.17

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



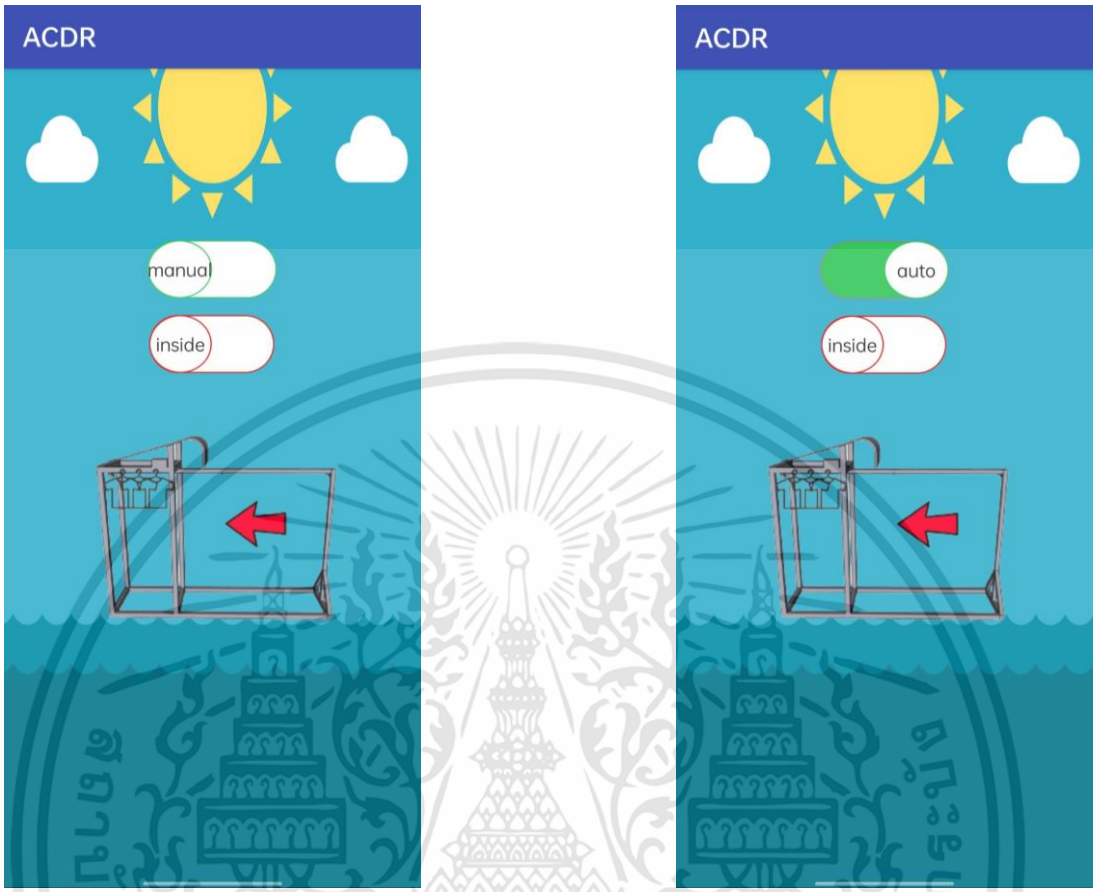
รูปที่ 3.17 หน้า DISPLAY ของแอปพลิเคชัน

3. หน้า CONTROL ของแอปพลิเคชัน

ทำการออกแบบส่วนต่อประสานผู้ใช้งานเพื่อให้ผู้ใช้งานสามารถเลือกควบคุมการทำงานของราวตากผ้าอัตโนมัติได้ 2 รูปแบบ ได้แก่ รูปแบบการทำงาน และรูปแบบการจัดเก็บของราวตากผ้า

รูปแบบที่ 1 คือ รูปแบบของโหมดการทำงาน โดยผู้ใช้งานสามารถเลือกควบคุมการทำงานของราวตากผ้าอัตโนมัติได้ 2 โหมด คือ การทำงานในโหมดอัตโนมัติ (auto) หมายถึง ให้ราวตากผ้าอัตโนมัติทำงานโดยประมวลผลจากข้อมูลต่าง ๆ ที่ได้จากเซ็นเซอร์ได้อย่างอัตโนมัติไม่รอรับคำสั่งจากแอปพลิเคชัน และการทำงานในโหมดการสั่งการด้วยมือ (manual) หมายถึง ให้ราวตากผ้าอัตโนมัติทำงานโดยรอรับคำสั่งจากแอปพลิเคชันเท่านั้น แสดงดังรูปที่ 3.18

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

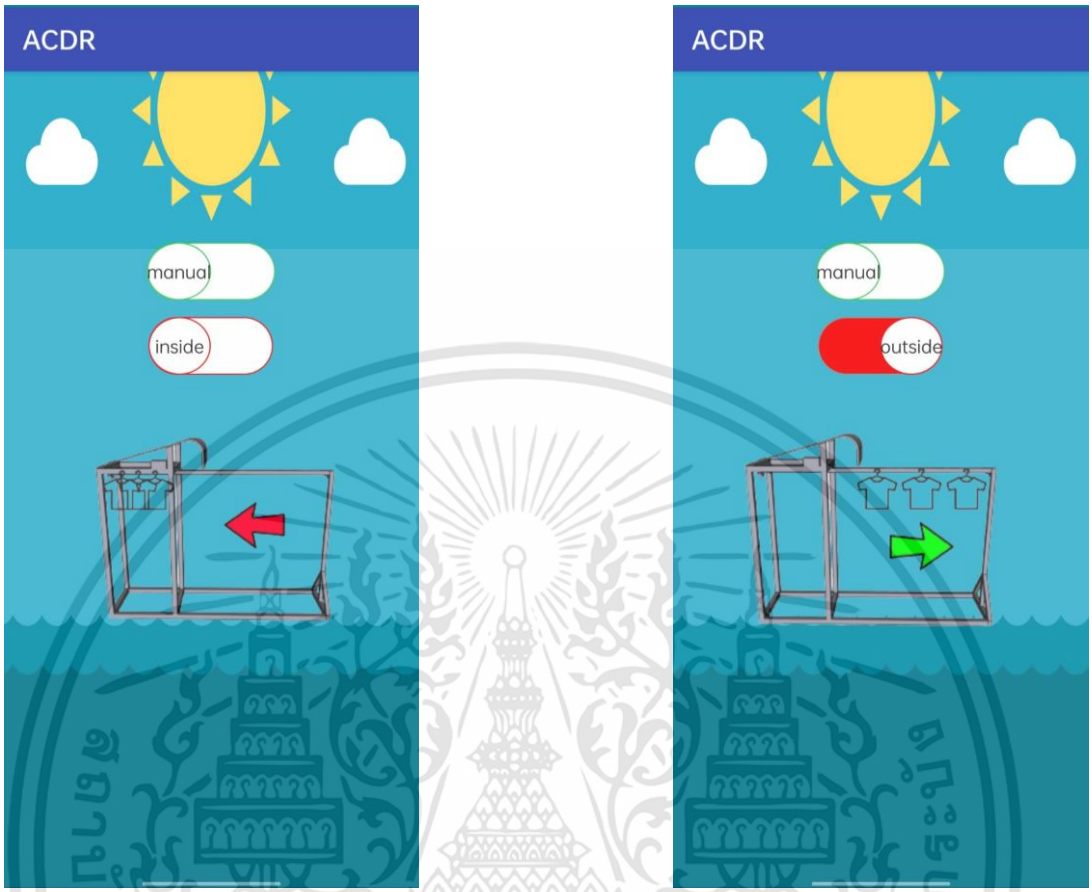


(ก) โหมด auto (ข) โหมด manual

รูปที่ 3.18 ตัวอย่างการควบคุมโหมดการทำงานของราวตากผ้าอัตโนมัติ

รูปแบบที่ 2 คือ รูปแบบของโหมดการควบคุมการจัดเก็บของราวตากผ้าอัตโนมัติ โดยผู้ใช้งานสามารถเลือกควบคุมการทำงานของราวตากผ้าอัตโนมัติได้สองโหมด คือ การทำงานในโหมดการควบคุมรอกให้อยู่ในตำแหน่งเก็บผ้า (inside) และการทำงานในโหมดการควบคุมรอกให้อยู่ในตำแหน่งตากผ้า (outside) แสดงดังรูปที่ 3.19

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(ก) inside (ข) outside

รูปที่ 3.19 ตัวอย่างโหมดการควบคุมการจัดเก็บของราวตากผ้าอัตโนมัติ

3.2 เครื่องมือที่ใช้ในการทดลอง

3.2.1 มอเตอร์

ในปริญญานิพนธ์นี้ผู้จัดทำใช้มอเตอร์กระจกไฟฟ้า 12 โวลต์ เพื่อทำหน้าที่ชักรอกเข้าและออก แสดงดังรูปที่ 2.4

3.2.2 โมดูลขับมอเตอร์

ในปริญญานิพนธ์นี้ผู้จัดทำใช้โมดูลขับมอเตอร์ L298N เพื่อควบคุมทิศทางของมอเตอร์ แสดงดังรูปที่ 2.3

3.2.3 ลิมิตสวิตช์

ในปริญญานิพนธ์นี้ผู้จัดทำใช้ลิมิตสวิตช์ LEMA กระแสไฟฟ้า16 แอมแปร์ แรงดันไฟฟ้า 250 โวลต์ เพื่อหยุดการทำงานของมอเตอร์ แสดงดังรูปที่ 2.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.4 บอร์ดไข่ปลา

ในปริญญานิพนธ์นี้ผู้จัดทำใช้บอร์ดไข่ปลา เพื่อเชื่อมต่อตัวต้านทาน และเพิ่มจุดเชื่อมต่อแรงดันไฟฟ้า 5 โวลต์ และ GND แสดงดังรูปที่ 2.6

3.2.5 แบตเตอรี่

ในปริญญานิพนธ์นี้ผู้จัดทำใช้แบตเตอรี่แรงดันไฟฟ้า 12 โวลต์ กระแสไฟฟ้า 2.16 แอมแปร์ เพื่อจ่ายไฟฟ้าให้โมดูลขับมอเตอร์ L298N แสดงดังรูปที่ 2.8

3.2.6 เซนเซอร์

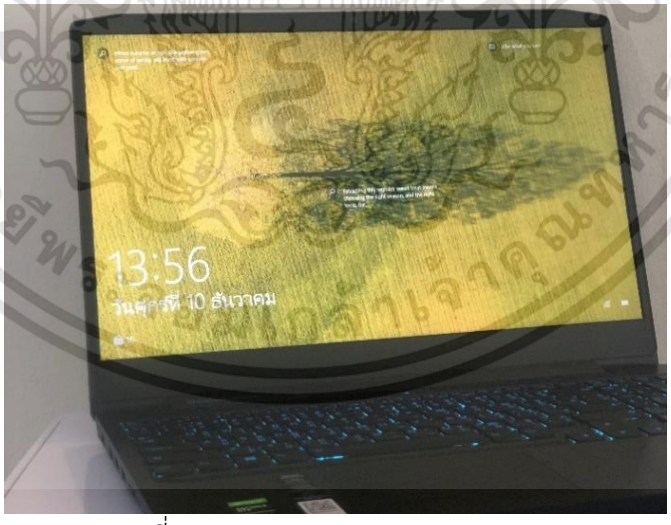
ในปริญญานิพนธ์นี้ผู้จัดทำใช้เซนเซอร์ ประกอบด้วย เซนเซอร์ตรวจวัดความเข้มแสง GY-2561 TSL 2561 โมดูลตรวจจับน้ำฝน และเซนเซอร์วัดความชื้นและอุณหภูมิ DHT 11 แสดงดังรูปที่ 2.14 รูปที่ 2.12 และรูปที่ 2.13 ตามลำดับ

3.2.7 Raspberry Pi

ในปริญญานิพนธ์นี้ผู้จัดทำใช้ Raspberry Pi 4 Model B เป็นหน่วยประมวลผลกลางสำหรับควบคุมการทำงานของอุปกรณ์ทั้งหมด แสดงดังรูปที่ 2.16

3.2.8 แล็ปท็อป

ในปริญญานิพนธ์นี้ผู้จัดทำใช้ Lenovo ideapadGAMING เพื่อป้อนชุดคำสั่งไปยัง Raspberry pi แสดงดังรูปที่ 3.20



รูปที่ 3.20 Lenovo IdeaPad Gaming

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.9 โทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์

ในปริญญานิพนธ์นี้ผู้จัดทำใช้ Realme6 Pro เพื่อใช้ในการควบคุมการทำงานของแอปพลิเคชัน แสดงดังรูปที่ 3.21

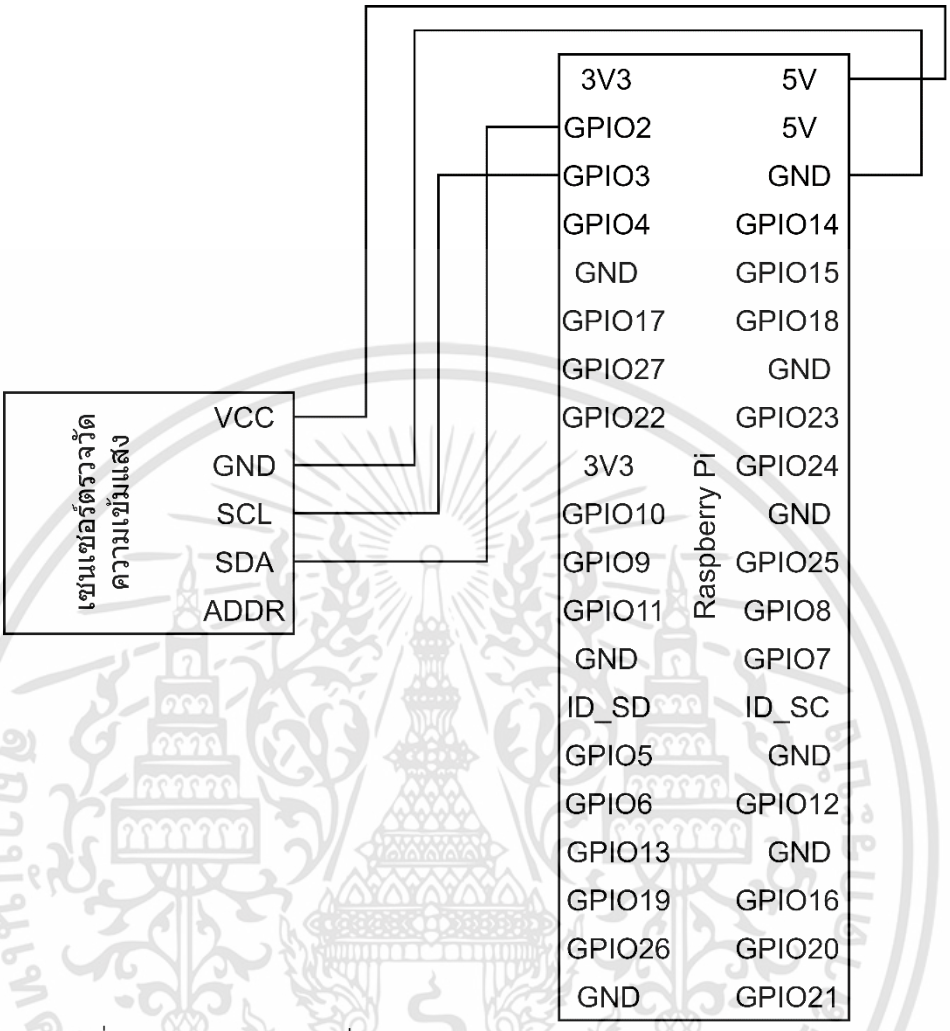
รูปที่ 3.21 Realme6 Pro

3.3 การจัดเก็บผลการทดลอง

3.3.1 ทดสอบการทำงานของเซนเซอร์ตรวจวัดความเข้มแสง GY-2561 TSL 2561

ทำการเชื่อมต่อสายไฟจากตำแหน่ง SCL SDA GND และ VCC ของเซนเซอร์ตรวจวัดความเข้มแสงไปยัง Raspberry Pi ดังรูปที่ 3.22 โดย SCL และ SDA คือตัวส่งข้อมูลที่ได้รับมาไปยัง Raspberry Pi GND คือแรงดันไฟฟ้าขั้วลบ และ VCC คือแรงดันไฟฟ้าขั้วบวก 5 โวลต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

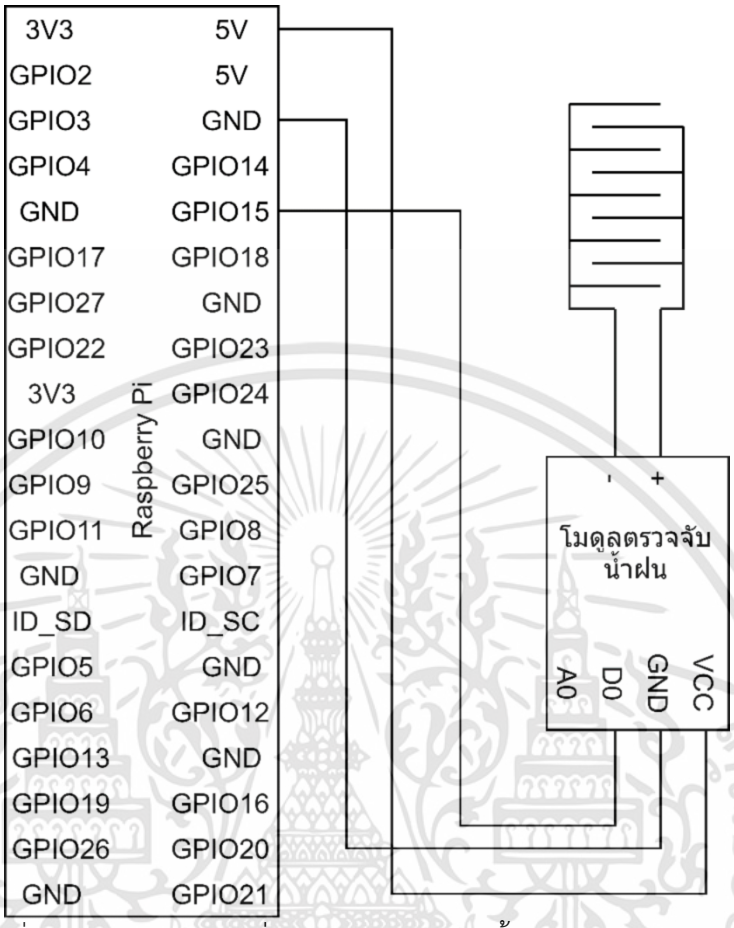


รูปที่ 3.22 แผนภาพการเชื่อมต่อเซนเซอร์วัดความเข้มแสงกับ Raspberry Pi

3.3.2 ทดสอบการทำงานของโมดูลตรวจจับน้ำฝน

ทำการเชื่อมต่อสายไฟจากตำแหน่ง D0 GND และ VCC ของโมดูลตรวจจับน้ำฝนไปยัง Raspberry Pi ดังรูปที่ 3.23 โดย D0 คือสัญญาณขาออกในรูปแบบดิจิทัล มีค่า 0 กับ 1 ต่อไปยังขา GPIO 15 GND คือแรงดันไฟฟ้าขั้วลบ และ VCC คือแรงดันไฟฟ้าขั้วบวก 5 โวลต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

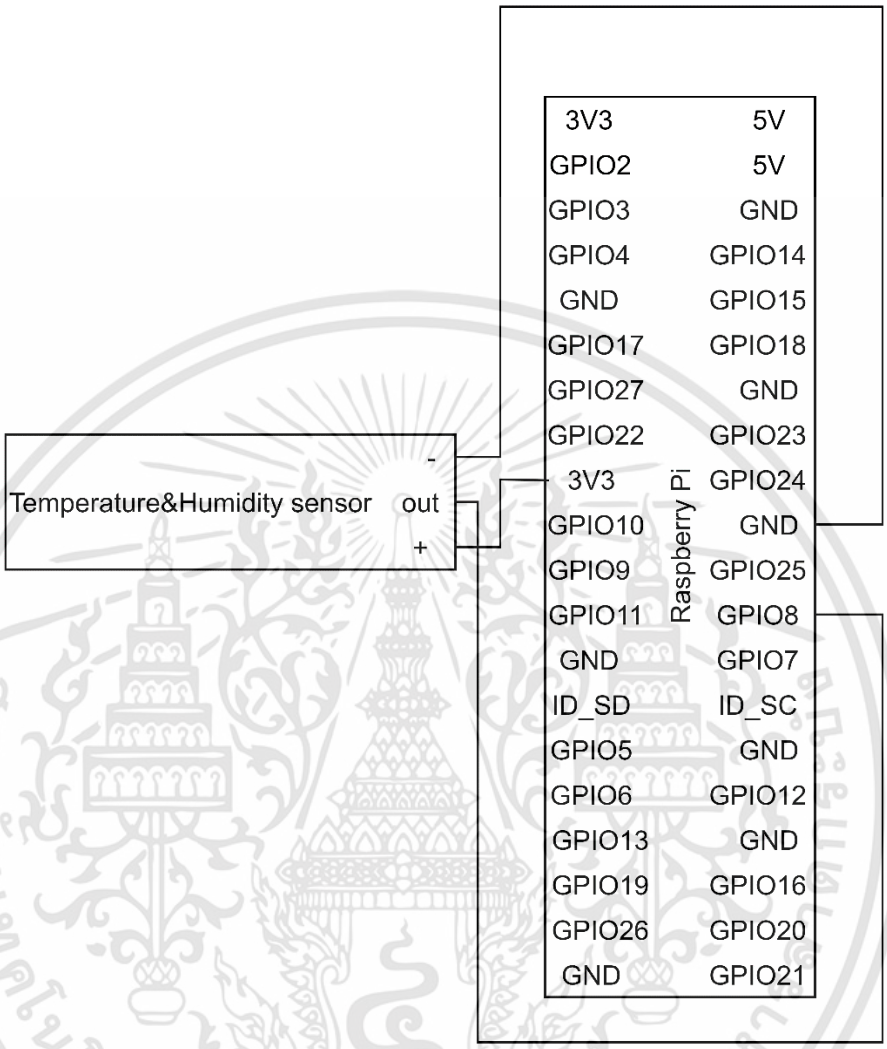


รูปที่ 3.23 แผนภาพการเชื่อมต่อโมดูลตรวจจับน้ำฝนกับ Raspberry Pi

3.3.3 ทดสอบการทำงานของเซนเซอร์วัดความชื้นและอุณหภูมิ DHT 22

ทำการเชื่อมต่อสายไฟจากตำแหน่ง OUT GND และ VCC ของเซนเซอร์ตรวจวัดความชื้นและอุณหภูมิไปยัง Raspberry Pi ดังรูปที่ 3.24 โดย OUT คือสัญญาณขาออก ต่อไปยังขา GPIO 21 GND คือแรงดันไฟฟ้าขั้วลบ และ VCC คือแรงดันไฟฟ้าขั้วบวก 5 โวลต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

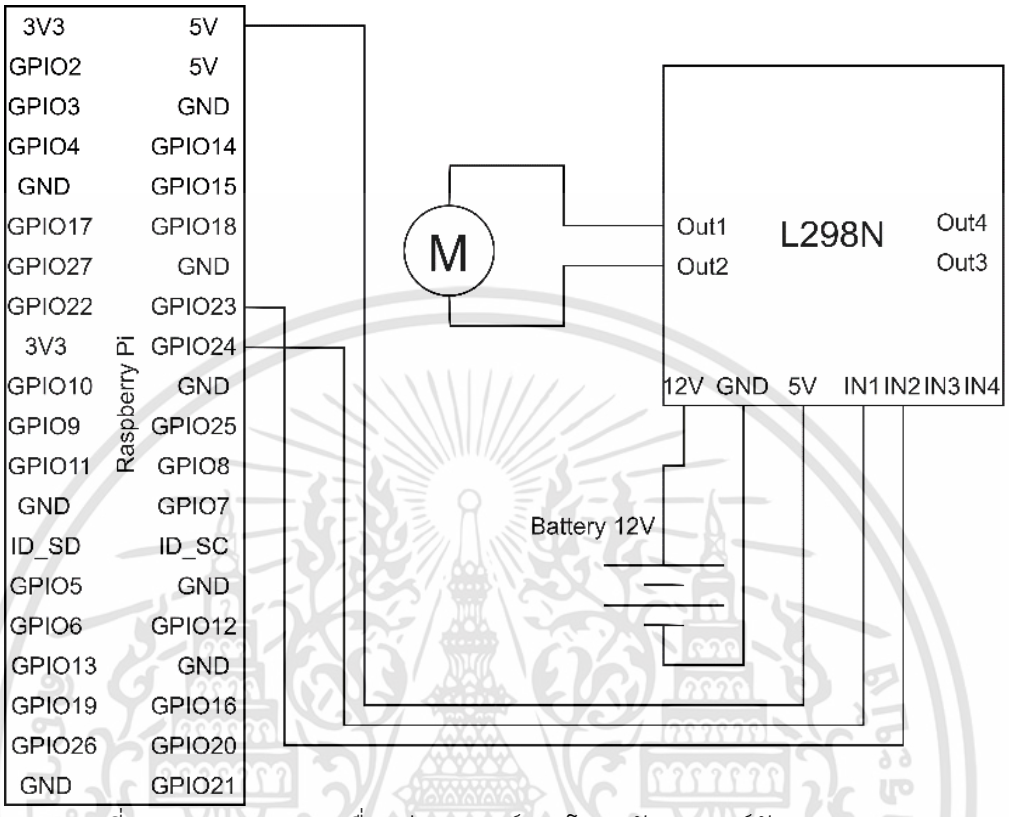


รูปที่ 3.24 การเชื่อมต่อเซนเซอร์วัดความชื้นและอุณหภูมิกับ Raspberry Pi

3.3.4 ทดสอบการทำงานของมอเตอร์

เป็นการทดสอบประสิทธิภาพการควบคุมความเร็วรอบและทิศทางการหมุนของมอเตอร์ โดยการทดสอบจะเป็นการเขียนชุดคำสั่งการทำงานด้วยภาษาไพธอน ลงบนบอร์ด Raspberry Pi ที่เชื่อมต่ออยู่กับโมดูลขับมอเตอร์ ทำการเชื่อมต่อสายไฟจากตำแหน่ง GND +5 โวลต์ In1 In2 และ ENA ของโมดูลขับมอเตอร์ไปที่ Raspberry Pi ดังรูปที่ 3.25 โดย GND คือแรงดันไฟฟ้าขั้วลบ และ +5V คือแรงดันไฟฟ้าขั้วบวกขนาด 5 โวลต์ In1 คือส่วนการควบคุมมอเตอร์ในทิศทางตามเข็มนาฬิกา In2 คือส่วนการควบคุมมอเตอร์ในทิศทางทวนเข็มนาฬิกา และ ENA คือส่วนการควบคุมแรงดันของมอเตอร์ จากนั้นเชื่อมต่อสายไฟจากตำแหน่ง OUT1 และ OUT2 ของโมดูลขับมอเตอร์ไปที่มอเตอร์ โดย OUT1 คือแรงดันไฟฟ้าขั้วบวก OUT2 คือแรงดันไฟฟ้าขั้วลบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.25 แผนภาพการเชื่อมต่อมอเตอร์และโมดูลขับมอเตอร์กับ Raspberry Pi

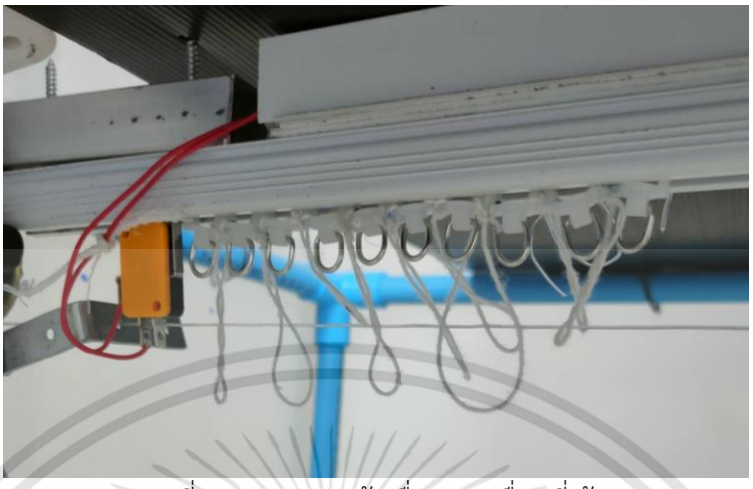
3.3.5 ทดสอบการทำงานตามสภาพภูมิอากาศ

ทดสอบการจำลองสภาพภูมิอากาศ โดยทดสอบ สวิตช์ ลิมิตสวิตช์ มอเตอร์ โมดูลขับมอเตอร์ โมดูลตรวจจับน้ำฝน และเซนเซอร์ตรวจวัดความเข้มแสง ว่าสามารถทำงานร่วมกันได้โดยเป็นไปตามเงื่อนไขที่กำหนดดังแผนผังในรูปที่ 3.11

3.3.6 ทดสอบระยะเวลาการเลื่อนรอก

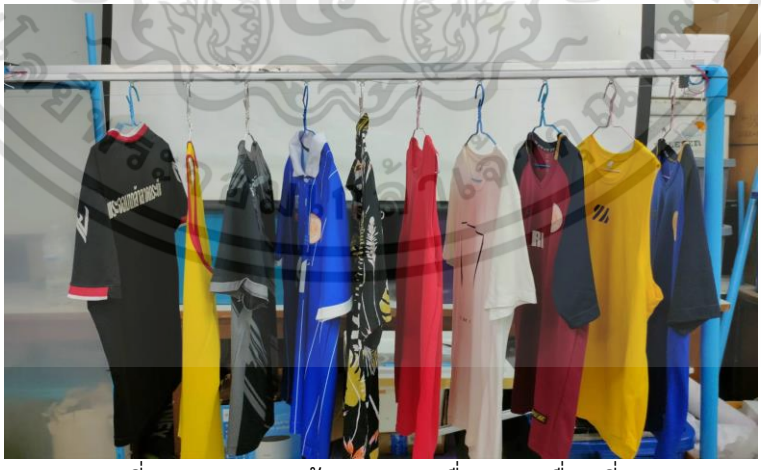
ทดสอบความสัมพันธ์ระหว่างเวลาในการเลื่อนของรอกตั้งแต่ด้านในออกไปด้านนอกจนกระทั้งชนกับลิมิตสวิตช์เมื่อผ้าแห้งและผ้าเปียก เพื่อสังเกตเวลาที่เปลี่ยนแปลงไป ทดสอบโดยการหมุนรอกเมื่อไม่แขวนผ้า แสดงดังรูปที่ 3.26 และ รูปที่ 3.27 และการหมุนรอกเมื่อแขวนผ้าเปียก แสดงดังรูปที่ 3.28 และ รูปที่ 3.29

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.26 ราวตากผ้าเมื่อรอกเคลื่อนที่เข้า

รูปที่ 3.27 ราวตากผ้าเมื่อรอกเคลื่อนที่ออก



รูปที่ 3.28 ราวตากผ้าขณะตากเมื่อรอกเคลื่อนที่ออก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.29 ราวตากผ้าขณะตากเมื่อรอกเคลื่อนที่เข้า

3.3.7 ทดสอบการรับส่งข้อมูลระหว่าง Raspberry Pi กับฐานข้อมูล Firebase Realtime Database

ทดสอบการรับส่งข้อมูลด้วยการเขียนโปรแกรมภาษาไพธอนบน Raspberry Pi เพื่อรับส่งข้อมูลต่าง ๆ บนฐานข้อมูล Firebase Realtime Database เพื่อเก็บข้อมูลของเซนเซอร์ที่เปลี่ยนแปลงไปในแต่ละช่วงเวลา หรือดึงข้อมูลกลับเพื่อควบคุมมอเตอร์

3.3.8 ทดสอบการเชื่อมต่อระหว่างฐานข้อมูล Firebase Realtime Database กับแอปพลิเคชัน

ทำการทดสอบด้วยการเขียนโปรแกรมภาษา Java สั่งการแอปพลิเคชันให้ทำการเชื่อมต่อกับฐานข้อมูล Firebase Realtime Database เพื่อแสดงผลข้อมูลที่ได้รับจากเซนเซอร์และส่งคำสั่งควบคุมอุปกรณ์ไปเก็บไว้บนฐานข้อมูล

3.3.9 ทดสอบการใช้งานแอปพลิเคชัน

ทำการทดสอบการใช้งานจริงของแอปพลิเคชันประกอบด้วยการทดสอบการแสดงผลข้อมูลของเซนเซอร์ตามเวลาที่เปลี่ยนแปลงไป และทดสอบการทำงานของคำสั่งควบคุมอุปกรณ์ของราวตากผ้าทั้งหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

ผลการทดลอง

4.1 ผลการทดสอบการทำงานของเซนเซอร์ตรวจวัดความเข้มแสง

ในการทดสอบการทำงานของเซนเซอร์ตรวจวัดความเข้มแสง ผู้จัดทำได้ทำการเก็บค่าความเข้มแสงทุก 1 ชั่วโมงตั้งแต่เวลา 6 นาฬิกา ถึงเวลา 18 นาฬิกา เป็นเวลา 3 วัน เพื่อสังเกตความสว่างที่เปลี่ยนแปลงไปในแต่ละช่วงเวลา แสดงดังตารางที่ 4.1

ตารางที่ 4.1 ค่าความเข้มแสงในทุก 1 ชั่วโมงตั้งแต่เวลา 6 นาฬิกา ถึงเวลา 18 นาฬิกา เป็นเวลา 3 วัน

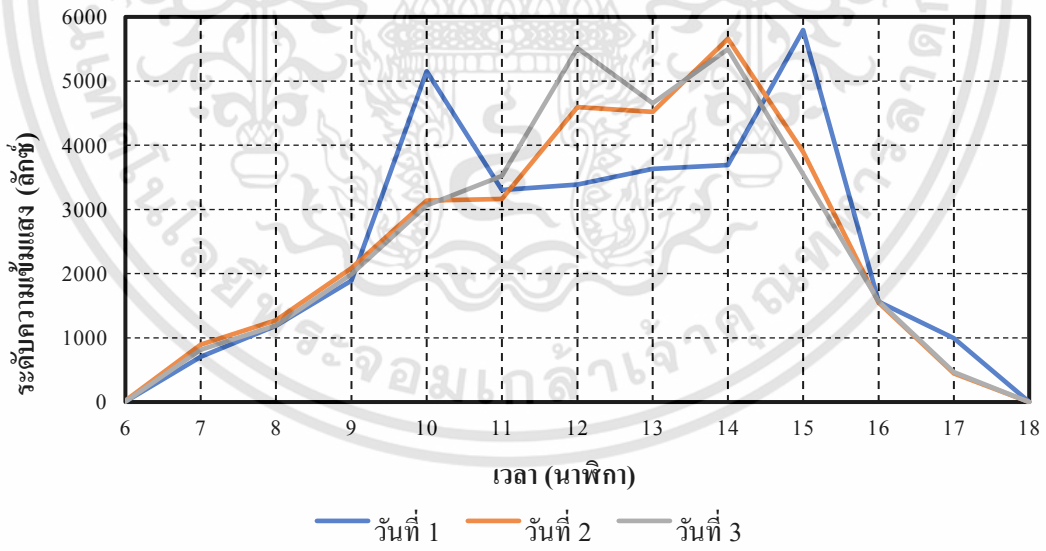
เวลา (นาฬิกา)	ค่าความเข้มแสง (ลักซ์)		
	วันที่ 1	วันที่ 2	วันที่ 3
6.00	6	19	7
7.00	705	892	816
8.00	1189	1273	1194
9.00	1891	2085	1991
10.00	5154	3136	3065
11.00	3301	3163	3521
12.00	3390	4593	5512
13.00	3631	4522	4655
14.00	3694	5659	5495
15.00	5795	3899	3549
16.00	1563	1549	1568

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.1 ค่าความเข้มแสงในทุก 1 ชั่วโมงตั้งแต่เวลา 6 นาฬิกา ถึงเวลา 18 นาฬิกา เป็นเวลา 3 วัน (ต่อ)

เวลา (นาฬิกา)	ค่าความเข้มแสง (ลักซ์)		
	วันที่ 1	วันที่ 2	วันที่ 3
17.00	994	444	465
18.00	2	0	0

จากตารางที่ 4.1 สามารถวิเคราะห์แนวโน้มความเข้มแสงจากเซนเซอร์ที่วัดได้ในช่วงเวลา 6 นาฬิกาถึง 18 นาฬิกา สังเกตว่า ค่าความเข้มแสงจะอยู่ที่ 0 ลักซ์ ถึง 5795 ลักซ์ โดยในช่วงเช้าถึงบ่ายมีแนวโน้มที่ความเข้มแสงจะสูงขึ้นและสูงสุดในช่วง 14 นาฬิกา ถึง 15 นาฬิกาของแต่ละวัน และในช่วงบ่ายถึงเย็นมีแนวโน้มที่ความเข้มแสงจะลดลงและต่ำสุดในเวลา 18 นาฬิกา ทั้งนี้ขึ้นอยู่กับสภาพอากาศที่มีการเปลี่ยนแปลงในแต่ละวันทำให้ช่วงเวลาเดียวกันมีค่าความเข้มแสงที่แตกต่างกัน ขึ้นลงในเวลาที่แตกต่างกันในแต่ละวัน



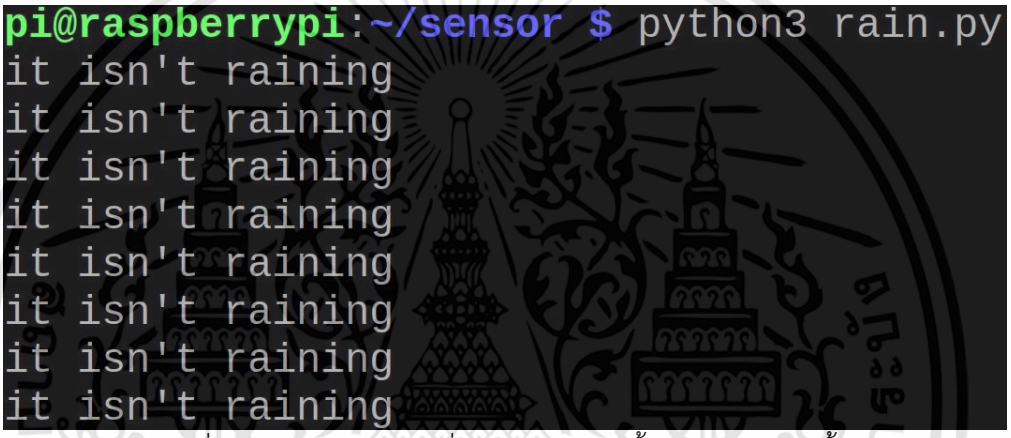
รูปที่ 4.1 ความเข้มแสงในช่วงเวลา 6 นาฬิกา ถึงเวลา 18 นาฬิกา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 ผลการทดสอบการทำงานของโมดูลตรวจจับน้ำฝน

ในการทดสอบการทำงานของโมดูลตรวจจับน้ำฝน ผู้จัดทำได้ทำการจำลองสภาพภูมิอากาศ คือ ใช้การหยดน้ำเปรียบเหมือนการที่เกิดฝนตก แล้วทำการอ่านค่าที่ได้เพื่อสังเกตความเปลี่ยนแปลง

ผลที่ได้คือ เมื่อเซนเซอร์สามารถตรวจพบน้ำฝนบริเวณโดยรอบ จะแสดงข้อความ “It raining” และเมื่อเซนเซอร์ตรวจไม่พบน้ำ จะแสดงข้อความ “It isn’t raining” แสดงดังรูปที่ 4.2 และ 4.3



รูปที่ 4.2 ผลการทดสอบเมื่อโมดูลตรวจจับน้ำฝนตรวจไม่พบน้ำ

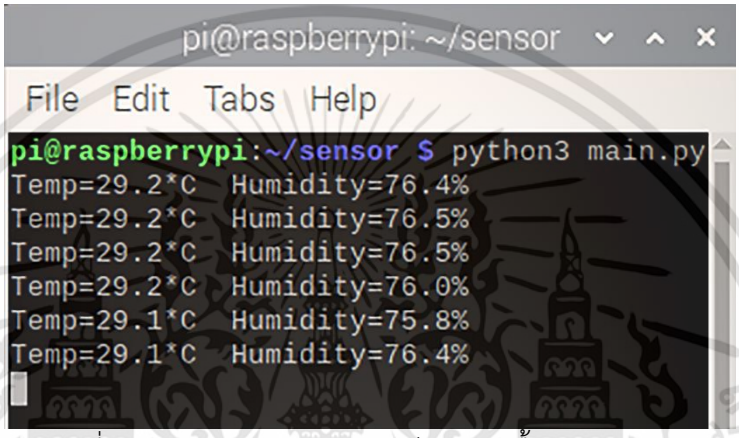
รูปที่ 4.3 ผลการทดสอบเมื่อโมดูลตรวจจับน้ำฝนตรวจพบน้ำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 ผลการทดสอบการทำงานของเซนเซอร์วัดความชื้นและอุณหภูมิ

ในการทดสอบการทำงานของเซนเซอร์วัดความชื้นและอุณหภูมิ ผู้จัดทำได้ทำการเก็บค่าความชื้นและอุณหภูมิเพื่อสังเกตความชื้นและอุณหภูมิที่เปลี่ยนแปลงไปในแต่ละช่วงเวลา

ผลที่ได้คือ เมื่อเซนเซอร์สามารถระบุความชื้นและอุณหภูมิบริเวณโดยรอบ จะแสดงข้อความ “Temp = อุณหภูมิ (℃) และ Humidity = ความชื้น (%)” แสดงดังรูปที่ 4.4



รูปที่ 4.4 ผลการทดสอบเซนเซอร์วัดความชื้นและอุณหภูมิ

4.4 ผลการทดสอบการทำงานมอเตอร์

ในการทดสอบการทำงานของมอเตอร์ ผู้จัดทำได้ทำการทดสอบการหมุนของมอเตอร์โดยจับเวลาในการเลื่อนของรอกจากด้านในไปถึงด้านนอกจนกระทั่งชนกับลิมิตสวิตช์ทั้งในขณะตากผ้าเปียก และไม่ตากผ้า เพื่อสังเกตความเร็วรอบในการหมุนของมอเตอร์และคำนวนประสิทธิภาพของมอเตอร์แสดงดังตารางที่ 4.2

ตารางที่ 4.2 ตารางทดสอบการหมุนของมอเตอร์

สถานะของราวตากผ้า	ความเร็วรอบ (รอบต่อนาที)	ประสิทธิภาพ (เปอร์เซ็นต์)
ไม่ตากผ้า	41.18	100
ตากผ้าเปียก	38.48	93.44

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากข้อมูลข้างต้นสรุปได้ว่า เมื่อไม่ตากผ้ามอเตอร์จะหมุนรอกจากด้านในไปถึงด้านนอกจนกระทั่งชนกับลิมิตสวิตช์ด้วยความเร็วรอบ 41.18 รอบต่อนาที คิดเป็นประสิทธิภาพ 100 เปอร์เซ็นต์ และเมื่อตากผ้าเปียกมอเตอร์จะหมุนรอกจากด้านในไปถึงด้านนอกจนกระทั่งชนกับลิมิตสวิตช์ด้วยความเร็วรอบ 38.48 รอบต่อนาที คิดเป็นประสิทธิภาพ 93.44 เปอร์เซ็นต์ กล่าวคือ เมื่อทำการตากผ้าเปียกจะส่งผลให้มอเตอร์มีความเร็วรอบที่ลดลงเนื่องจากน้ำหนักของผ้าที่นำมาตาก แต่ยังมีประสิทธิภาพที่มากพอในการใช้งาน

4.5 ผลการทดสอบการทำงานตามสภาพภูมิอากาศ

ในการทดสอบการทำงานตามสภาพภูมิอากาศ ผู้จัดทำได้ทำการทดสอบการหมุนของมอเตอร์ว่าเป็นไปตามเงื่อนไขที่กำหนดหรือไม่ โดยเงื่อนไขที่กำหนดแสดงดังรูปที่ 3.11 โดยผลการทดสอบการหมุนแสดงดังตารางที่ 4.3

ตารางที่ 4.3 ผลการทดสอบการหมุนของมอเตอร์

ความเข้มแสง (ลักซ์)	สถานะน้ำฝน	ผลการทดสอบ
0	มีน้ำฝน	มอเตอร์เลื่อนเข้าด้านใน
0	มีน้ำฝน	มอเตอร์เลื่อนเข้าด้านใน
มากกว่า 0	ไม่มีน้ำฝน	มอเตอร์เลื่อนเข้าด้านใน
มากกว่า 0	ไม่มีน้ำฝน	มอเตอร์เลื่อนออกด้านนอก

จากผลการทดสอบข้างต้นสรุปได้ว่า การหมุนของมอเตอร์เป็นไปตามเงื่อนไขที่กำหนด โดยเมื่อความเข้มแสงเท่ากับ 0 ลักซ์ และมีน้ำฝน มอเตอร์จะเลื่อนเข้าด้านใน เมื่อความเข้มแสงเท่ากับ 0 ลักซ์ และมีน้ำฝน มอเตอร์จะเลื่อนเข้าด้านใน เมื่อความเข้มแสงมากกว่า 0 ลักซ์ และไม่มีน้ำฝน มอเตอร์จะเลื่อนเข้าด้านใน และเมื่อความเข้มแสงมากกว่า 0 ลักซ์ และไม่มีน้ำฝน มอเตอร์จะเลื่อนออกด้านนอก

4.6 ผลการทดสอบการเลื่อนของรอก

ในการทดสอบการเลื่อนของรอก ผู้จัดทำได้ทำการเปรียบเทียบเวลาในการเลื่อนของรอกตั้งแต่ด้านในออกไปด้านนอกจนกระทั่งชนกับลิมิตสวิตช์เมื่อไม่มีผ้า และมีผ้าเปียกทั้งหมด 3 ครั้ง เพื่อสังเกตเวลาที่เปลี่ยนแปลงไป ค่าเฉลี่ย และส่วนเบี่ยงเบนมาตรฐานแสดงดังตารางที่ 4.4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.4 เปรียบเทียบเวลาการเลื่อนของรอกเมื่อไม่มีผ้า และมีผ้าเปียก

สถานะของผ้า	เวลา (วินาที)			ค่าเฉลี่ย (วินาที)	ส่วนเบี่ยงเบนมาตรฐาน
	ครั้งที่ 1	ครั้งที่ 2	ครั้งที่ 3		
ไม่มีผ้า	24.15	23.48	23.89	23.84	0.28
มีผ้าเปียก	25.26	25.69	25.91	25.62	0.27

จากข้อมูลข้างต้นสรุปได้ว่า การเลื่อนของรอกขณะตากผ้าเปียกใช้เวลามากกว่าการเลื่อนของรอกขณะไม่มีผ้าเป็นเวลา 1.78 วินาที โดยการเลื่อนของรอกตั้งแต่ด้านในออกไปด้านนอกจนกระทั่งชนกับลิมิตสวิตช์เมื่อไม่มีผ้าใช้เวลาเฉลี่ย 23.84 วินาที คิดเป็นส่วนเบี่ยงเบนมาตรฐาน 0.28 และการเลื่อนของรอกตั้งแต่ด้านในออกไปด้านนอกจนกระทั่งชนกับลิมิตสวิตช์เมื่อผ้าเปียกใช้เวลาเฉลี่ย 25.62 วินาที คิดเป็นส่วนเบี่ยงเบนมาตรฐาน 0.27

4.7 ผลการทดสอบการรับส่งข้อมูลระหว่าง Raspberry Pi กับฐานข้อมูล Firebase Realtime Database

ผู้จัดทำได้ออกแบบชุดคำสั่งการรับส่งข้อมูลและการเชื่อมต่อกับฐานข้อมูล Firebase Realtime Database โดยสร้างโครงงานทดลองที่มีชื่อว่า testfire เพื่อทำการรับข้อมูลจาก Raspberry Pi โดยได้สร้างที่อยู่ของไฟล์ไว้สำหรับเชื่อมต่อข้อมูลจากเซนเซอร์ต่าง ๆ เพื่อแสดงบนฐานข้อมูล และที่อยู่ของไฟล์สำหรับการควบคุมมอเตอร์ ซึ่งจะมีชื่อที่อยู่ของไฟล์แทนคำสั่งทั้งหมดดังต่อไปนี้ แสดงดังรูปที่ 4.5

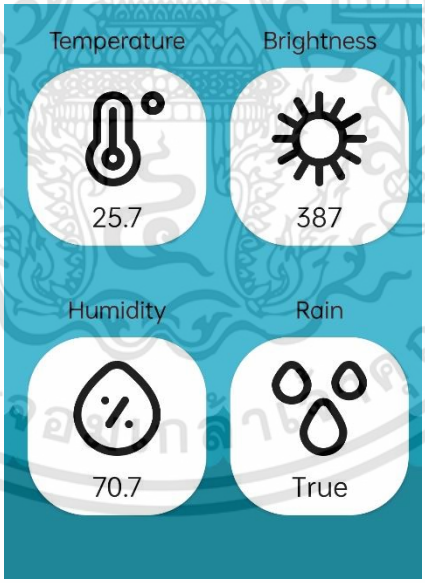
รูปที่ 4.5 ผลการทดสอบชื่อที่อยู่ของไฟล์ที่เชื่อมต่อระหว่างอุปกรณ์และ Firebase Realtime Database

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากการทดสอบการรับส่งข้อมูลระหว่าง Raspberry Pi กับฐานข้อมูล Firebase Realtime Database สามารถสรุปได้ว่า Raspberry Pi สามารถส่งคำสั่งการสร้างชื่อที่อยู่ของไฟล์ไปยังฐานข้อมูลได้ โดยจะเห็นว่าค่าที่ได้รับจากเซนเซอร์และสถานะการทำงานต่าง ๆ จะแสดงในส่วนของไฟล์ชุดคำสั่งที่สร้างไว้ และมีการรายงานค่าทันทีเมื่อข้อมูลมีการเปลี่ยนแปลง และยังสามารถรับคำสั่งการทำงานจากการสั่งงานแอปพลิเคชันผ่านฐานข้อมูลได้

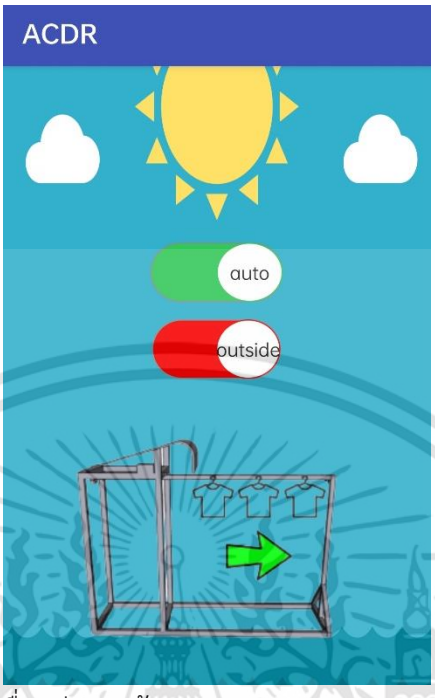
4.8 ผลการทดสอบการเชื่อมต่อระหว่างฐานข้อมูล Firebase Realtime Database กับแอปพลิเคชัน

ผู้จัดทำได้ออกแบบชุดคำสั่งการรับส่งข้อมูลและการเชื่อมต่อระหว่างแอปพลิเคชันบน Android กับฐานข้อมูล Firebase Realtime Database เพื่อดูข้อมูลอุณหภูมิ ความชื้น ความเข้มแสง และสถานะของน้ำฝนที่เก็บอยู่ในฐานข้อมูลผ่านแอปพลิเคชันบน Android ได้ตามเวลาจริงในหน้า Display แสดงดังรูปที่ 4.6 และเพื่อเลือกโหมดการทำงานสำหรับการควบคุมการทำงานของราวตากผ้าได้ทั้งในโหมดอัตโนมัติ และโหมดสั่งการด้วยมือ รวมไปถึงการควบคุมรอกให้อยู่ในตำแหน่ง inside หรือ outside ในหน้า Control แสดงดังรูปที่ 4.7



รูปที่ 4.6 ผลการทดสอบการเชื่อมต่อฐานข้อมูล Firebase Realtime Database กับแอปพลิเคชันในหน้า Display

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.7 ผลการทดสอบการเชื่อมต่อฐานข้อมูล Firebase Realtime Database กับแอปพลิเคชันในหน้า Control

จากการทดสอบสามารถสรุปได้ว่า แอปพลิเคชันบน Android สามารถรับเชื่อมต่อกับฐานข้อมูล Firebase Realtime Database ได้ โดยจะเห็นว่าในหน้า Display ของแอปพลิเคชันสามารถแสดงค่าที่ได้รับจากเซนเซอร์ต่าง ๆ ได้ตามค่าที่อยู่บนฐานข้อมูล ในหน้า Control สามารถควบคุมการทำงานในแต่ละโหมด และควบคุมรอกได้เช่นกัน

4.9 ผลการทดสอบการใช้งานแอปพลิเคชัน

ในการทดสอบการใช้งานแอปพลิเคชันจะทดการทดสอบทั้งหมด 2 รูปแบบ คือ ทดสอบการแสดงผลข้อมูลของเซนเซอร์ตามเวลาที่เปลี่ยนแปลงไป และทดสอบการควบคุมรอก

4.9.1 ทดสอบการแสดงผลข้อมูลของเซนเซอร์ตามเวลาที่เปลี่ยนแปลงไป

การแสดงผลข้อมูลขอเซนเซอร์ ผู้จัดทำได้ทำการเก็บข้อมูลความเข้มแสง น้ำฝน อุณหภูมิ และความชื้นที่แสดงบนแอปพลิเคชันทุก 1 ชั่วโมงตั้งแต่เวลา 6 นาฬิกา ถึงเวลา 18 นาฬิกา เป็นเวลา 3 วัน เพื่อสังเกตการเปลี่ยนแปลงไปในแต่ละช่วงเวลา แสดงดังตารางที่ 4.5 ถึง 4.7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.5 การแสดงผลข้อมูลของเซนเซอร์ทุก 1 ชั่วโมงตั้งแต่เวลา 6 นาฬิกา ถึงเวลา 18 นาฬิกาในวันที่ 1

เวลา (นาฬิกา)	เซนเซอร์			
	ความเข้มแสง (ลักซ์)	สถานะของน้ำฝน	อุณหภูมิ (องศาเซลเซียส)	ความชื้น (เปอร์เซ็นต์)
6	8	ไม่มีฝน	25.6	57.1
7	716	ไม่มีฝน	25.8	57.1
8	1190	ไม่มีฝน	26.2	57.0
9	1900	ไม่มีฝน	26.4	56.8
10	5050	ไม่มีฝน	27.8	56.7
11	3388	ไม่มีฝน	28.4	56.6
12	4580	ไม่มีฝน	29.0	56.4
13	5550	ไม่มีฝน	29.4	56.4
14	8206	ไม่มีฝน	29.5	56.4
15	5637	ไม่มีฝน	29.8	56.3
16	2687	ไม่มีฝน	28.6	56.6
17	1125	ไม่มีฝน	27.9	56.7
18	384	ไม่มีฝน	27.5	56.7

ตารางที่ 4.6 การแสดงผลข้อมูลของเซนเซอร์ทุก 1 ชั่วโมงตั้งแต่เวลา 6 นาฬิกา ถึงเวลา 18 นาฬิกาในวันที่ 2

เวลา (นาฬิกา)	เซนเซอร์			
	ความเข้มแสง (ลักซ์)	สถานะของน้ำฝน	อุณหภูมิ (องศาเซลเซียส)	ความชื้น (เปอร์เซ็นต์)
6	27	ไม่มีฝน	25.8	61.3
7	733	ไม่มีฝน	26.0	61.9
8	2033	ไม่มีฝน	26.3	62.0
9	2892	ไม่มีฝน	26.5	62.4
10	3624	ไม่มีฝน	28.0	62.5
11	4880	ไม่มีฝน	28.5	62.5
12	6019	ไม่มีฝน	28.7	62.6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.6 การแสดงผลข้อมูลของเซนเซอร์ทุก 1 ชั่วโมงตั้งแต่เวลา 6 นาฬิกา ถึงเวลา 18 นาฬิกาในวันที่ 2 (ต่อ)

เวลา (นาฬิกา)	เซนเซอร์			
	ความเข้มแสง (ลักซ์)	สถานะของน้ำฝน	อุณหภูมิ (องศาเซลเซียส)	ความชื้น (เปอร์เซ็นต์)
13	1449	มีฝน	27.0	61.2
14	1416	มีฝน	26.2	61.4
15	2357	ไม่มีฝน	26.0	61.5
16	2428	ไม่มีฝน	26.1	61.4
17	1641	ไม่มีฝน	26.0	61.5
18	398	ไม่มีฝน	25.9	61.5

ตารางที่ 4.7 การแสดงผลข้อมูลของเซนเซอร์ทุก 1 ชั่วโมงตั้งแต่เวลา 6 นาฬิกา ถึงเวลา 18 นาฬิกา ในวันที่ 3

เวลา (นาฬิกา)	เซนเซอร์			
	ความเข้มแสง (ลักซ์)	สถานะของน้ำฝน	ความเข้มแสง (ลักซ์)	ความชื้น
6	29	ไม่มีฝน	25.7	69.8
7	384	ไม่มีฝน	25.9	69.7
8	1357	ไม่มีฝน	26.2	69.8
9	2439	ไม่มีฝน	26.5	69.9
10	4965	ไม่มีฝน	27.5	70.1
11	2455	ไม่มีฝน	27.8	70.7
12	740	มีฝน	28.1	70.4
13	345	มีฝน	27.7	70.1
14	540	มีฝน	27.0	70.1
15	1464	มีฝน	26.9	70.0
16	1899	ไม่มีฝน	26.9	69.7
17	970	ไม่มีฝน	26.7	69.6
18	212	ไม่มีฝน	26.6	69.6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.9.2 ทดสอบการควบคุมรอก

ผู้จัดทำได้ทำการควบคุมราวตากผ้าผ่านแอปพลิเคชัน ในโหมดอัตโนมัติจะเป็นการทำงานตามสภาพภูมิอากาศ ส่วนในโหมดสั่งการด้วยมือ จะเป็นการทำงาน 2 รูปแบบ คือ การควบคุมรอกให้อยู่ในตำแหน่งเก็บผ้า (inside) และการควบคุมรอกให้อยู่ในตำแหน่งตากผ้า (outside) แสดงดังรูปที่ 4.8 และ 4.9

รูปที่ 4.8 ผลการควบคุมรอกให้อยู่ในตำแหน่งเก็บผ้า (inside)

รูปที่ 4.9 ผลการควบคุมรอกให้อยู่ในตำแหน่งตากผ้า (outside)

ผลการทดสอบ พบว่า ผู้ใช้งานสามารถสั่งการควบคุมการทำงานของรอกในระยะไกลโดยผ่านแอปพลิเคชันตามวัตถุประสงค์ของโครงงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปผลและข้อเสนอแนะ

5.1 สรุปผล

ปริญญานิพนธ์นี้เป็นการออกแบบราวตากผ้าอัตโนมัติที่สามารถตากและเก็บผ้าได้ด้วยระบบอัตโนมัติจากเซนเซอร์ตรวจวัดความเข้มแสง โมดูลตรวจจับน้ำฝน และเซนเซอร์ตรวจวัดความชื้นและอุณหภูมิทำงานร่วมกับมอเตอร์ และ Raspberry Pi เพื่อให้ผู้ใช้งานไม่จำเป็นต้องสังเกตสภาพภูมิอากาศตลอดเวลาเมื่อทำการตากผ้า สามารถควบคุมการตากผ้าได้จากระยะไกล และยังติดตามสภาพภูมิอากาศได้ผ่านแอปพลิเคชันบนโทรศัพท์มือถือระบบปฏิบัติการ Android

ในภาคการศึกษาที่ 1 ผู้จัดทำได้ออกแบบและจัดทำราวตากผ้าอัตโนมัติที่ประกอบไปด้วยการออกแบบและจัดทำโครงสร้างของชิ้นงาน ทดสอบการทำงานการทำงานของอุปกรณ์ต่าง ๆ พร้อมกับบันทึกผลเบื้องต้นได้แก่ ทดสอบการเลื่อนของรอก ทดสอบการทำงานของเซนเซอร์ ทดสอบการทำงานของมอเตอร์ และทดสอบการทำงานตามสภาพภูมิอากาศ พบว่าอุปกรณ์ทั้งหมดสามารถทำงานร่วมกันได้อย่างสมบูรณ์ส่งผลให้ราวตากผ้าสามารถทำงานได้อัตโนมัติตามเงื่อนไขที่กำหนด

โดยในภาคการศึกษาที่ 2 ผู้จัดทำได้ออกแบบและจัดทำราวตากผ้าอัตโนมัติที่ประกอบไปด้วยการออกแบบชุดคำสั่งการรับส่งข้อมูลของเซนเซอร์ต่าง ๆ ไปยังฐานข้อมูลตามเวลาจริง การออกแบบแอปพลิเคชันสำหรับควบคุมราวตากผ้าอัตโนมัติ และทดสอบการทำงานในภาพรวมทั้งหมด โดยเมื่อนำส่วนที่ได้ออกแบบทั้งหมดในภาคการศึกษาที่ 2 ไปใช้งานร่วมกันกับที่ออกแบบในภาคการศึกษาที่ 1 พบว่าการออกแบบและทดสอบทั้งหมดสามารถทำงานร่วมกันได้อย่างสมบูรณ์ส่งผลให้ราวตากผ้าสามารถทำงานได้อัตโนมัติตามวัตถุประสงค์ที่กำหนด

5.2 ข้อเสนอแนะ

ราวตากผ้ายังมีข้อจำกัดในเรื่องของขนาดของราวตากผ้าที่มีขนาดค่อนข้างใหญ่ ใช้พื้นที่ค่อนข้างมาก ส่งผลให้การใช้งานจริง ผู้ใช้งานจำเป็นต้องมีพื้นที่ที่มากเพียงพอต่อการติดตั้ง และเคลื่อนย้ายราวตากผ้าอัตโนมัติ โดยปกติต้องใช้วิธีการยกในการเคลื่อนย้าย ส่งผลต่อความสะดวกสบายของผู้ใช้งาน นอกจากนี้ยังมีข้อจำกัดเรื่องความแข็งแรงของโครงสร้างราวตากผ้าที่ใช้ท่อพีวีซีเป็นโครงสร้างหลัก หากใช้กลางแดดเป็นประจำ อาจส่งผลให้โครงสร้างเปราะแตกได้ มีข้อจำกัดเรื่องสภาพภูมิอากาศช่วงหลังฝนหยุดตกนั้นมีละอองน้ำติดอยู่ที่โมดูลตรวจจับน้ำฝน ทำให้การประมวลผลบนแอปพลิเคชันไม่สามารถนำผ้าออกไปตากได้อัตโนมัติ ทั้งนี้โครงสร้างชิ้นงานดังกล่าวผู้จัดทำได้เสนอรูปแบบของชิ้นงานต้นแบบซึ่งสามารถปรับเปลี่ยนและพัฒนาให้เป็นวัสดุที่แข็งแรงทนทานต่อสภาพแวดล้อมที่ดีขึ้นได้ตามงบประมาณในอนาคต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

[1] Matana Wiboonyasake. “ทำความรู้จักกับ Internet of Things.”
https://www.aware.co.th/iot-คืออะไร.

[2] จีระพงษ์ โพพันธุ์. “IoT ในชีวิตประจำวัน.”
https://kru-it.com/computing-science-m3/iot.Arduino.MakerZone.

[3] Cybertice. “L298N motor drive module.”
https://www.cybertice.com/product/42/l298n-motor-drive-module

[4] Wikipedia. “มอเตอร์ไฟฟ้า”
https://th.wikipedia.org/wiki/มอเตอร์.

[5] Google Sites. “ข้อมูลมอเตอร์ไฟฟ้ากระแสตรง.”
สืบค้นจาก: https://sites.google.com/site/khormulmorter.

PoundXi. “Raspberry Pi คืออะไร.”
https://poundxi.com/raspberry-pi-คืออะไร.

[6] Omron. “Snap Action Roller Lever Limit Switch ลิมิตสวิตช์ชนิดก้านบานพับ.”
https://th.rs-online.com/web/p/limit-switches/3860423/.

[7] Commandronestore. “PCB บอร์ดไข่ปลา 5x7 cm.”
https://commandronestore.com/products/bd002.php.

[8] PSP TECH. “ตัวต้านทาน(resister).”

[9] บริษัท ไตรเทพ อินดัสทรี จำกัด. “แบตเตอรี่คืออะไร.”
http://www.diy-solarcell.com/สาระน่ารู้พลังงาน%20(ความหมายของพลังงาน)แบตเตอรี่%20(Battery).html.

Joom. “100 X 1/4W 250V 300 Ohm 5% Axial Carbon Film Resistors.”
https://www.joom.com/en/products/1510568390541770105-191-1-26193-1243241460.

[10] บริษัท ป.กวิน จำกัด. “คุณสมบัติของท่อ PVC.”
https://www.pokawin.com/ 17433534/คุณสมบัติของท่อ-pvc.

[11] บริษัท ไชยเจริญเทค จำกัด. “อะคริลิคกับ คุณสมบัติเด่น และการนำมาประยุกต์ใช้งานอย่างเหมาะสม.”

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

https://www.chi.co.th/ article/article-861/.

[12] หลังคาเย็นสยาม (เทพารักษ์). “แผ่นหลังคาเมทัลชีท (Metal Sheet).”
https://www.pucoolroof .com/metal-sheet/.

[13] MLT. “โมดูลตรวจจับน้ำฝน.”
http://www.mltelectronic.com/โมดูลตรวจจับน้ำฝน--(Raindrop-Detection-Sensor-Module).

[14] lnwshop. “การใช้งาน DHT11.”
https://www.arduitronics.com/article/13/การใช้งาน-dht11-humitdity-and-temperature-sensor-กับบอร์ด-arduino.

[15] AllNewStep. “เซนเซอร์วัดแสงช่วงกว้าง GY-2561 TSL2561 เซนเซอร์แสง.”
https://www.allnewstep.com/ product/629/gy-2561-tsl2561-เซนเซอร์วัดแสงช่วงกว้าง-gy-2561-tsl2561-เซนเซอร์แสง.

[16] บริษัท สยาม โกลบอล กรุ๊ป จำกัด. “สวิตซ์เปิด-ปิด รุ่น JD03-A1-A05-108 SUMO.”
https://www.sgb.co.th/สวิตซ์-สวิตซ์เปิด-ปิด-รุ่น-jd03-a1-a05-108-สวิตช์- 4-ขา-มีไฟ-led-สวิตซ์ไฟ-สวิตช์.

[17] ArduinoPro. “Raspberry Pi Tutorial [EP1] : Raspberry Pi คืออะไร ?.”
http://www.arduino-makerzone.com/article/57/raspberry-pi-tutorial-ep1-raspberry-pi-คืออะไร.

[18] Choonewza. “ระบบปฏิบัติการ Raspbian.”
https://choonewza.medium.comการติดตั้งและตั้งค่าระบบ-raspberry-pi-3-model-b-fadd88dbfac0.

[19] Wannaphong. “ประวัติความเป็นมาของ Python.”
https://python3.wannaphong.com/2017/09/python.html.

[20] ชุติภรณ์ วิเศษชาติ. “Android คือ อะไร”
https://sites.google.com/a.thantong.ac.th/rabb-ptibati-kar-android/android-khux-xari.

[21] Palm’s. “เริ่มต้นสร้าง Android Application พื้นฐานด้วย Android Studio”
https://medium.com/@palmz/เริ่มต้นสร้าง-android-application-พื้นฐานด้วย-android-studio-lab-3sb04-3fda43b07a1.

[22] Android Developers. “User Interface & Navigation”
https://developer.android.com/guide/topics/ui.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

[23] “XML คืออะไร”

https://www.mindphp.com/คู่มือ/73-คืออะไร/2129-xml-คืออะไร.html.

[24] mr.win – TC Admin. “Java คืออะไร”

https://www.mindphp.com/คู่มือ/73-คืออะไร/2129-xml-คืออะไร.html.

[25] “firebase-คืออะไร”

https://www.4xtreme.com/2020/11/20/firebase-คืออะไร

[26] “ทำความรู้จัก firebase”

https://medium.com/@sirawit/firebase-คืออะไร-ทำความรู้จัก-firebase-ในช่วงต้นปี-2019-กัน-473a8e8699fb

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก

ชุดคำสั่งการรับข้อมูลจากเซนเซอร์บน Raspberry Pi

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนของเซนเซอร์ตรวจวัดความเข้มแสง

```
import smbus
import time

# Get I2C bus
bus = smbus.SMBus(1)

# TSL2561 address, 0x39(57)
# Select control register, 0x00(00) with command register, 0x80(128)
# 0x03(03)        Power ON mode
bus.write_byte_data(0x39, 0x00 | 0x80, 0x03)
# TSL2561 address, 0x39(57)
# Select timing register, 0x01(01) with command register, 0x80(128)
# 0x02(02)        Nominal integration time = 402ms
bus.write_byte_data(0x39, 0x01 | 0x80, 0x02)

time.sleep(0.5)

while True:
  data = bus.read_i2c_block_data(0x39, 0x0C | 0x80, 2)
  data1 = bus.read_i2c_block_data(0x39, 0x0E | 0x80, 2)

  # Convert the data
   ch0 = data[1] * 256 + data[0]
   ch1 = data1[1] * 256 + data1[0]

   print ("Full Spectrum(IR + Visible) :%d lux" %ch0)
   print ("Infrared Value :%d lux" %ch1)
   print ("Visible Value :%d lux\n" %(ch0 - ch1))
   time.sleep(2)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนของโมดูลตรวจจับน้ำฝน

```
import time
from time import sleep
from gpiozero import InputDevice

norain = InputDevice(15)

while True:
  if not norain.is_active:
    print ('raining')
  time.sleep(1)

  if norain.is_active:
   print ('no rain')
  time.sleep(1)
```

ส่วนของเซนเซอร์วัดความชื้นและอุณหภูมิ 1

```
import Adafruit_DHT as DHT

class DHT22(object):

    def __init__(self,out=None, units = "c"):

        self.sensor = None

        self.units = units

        self.out = out

        self.sensor = DHT.DHT22
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    def get_value(self):
        humidity, temp = self.read()
        temperature = getattr(self, "pass_" + self.units)(temp)
        return humidity, temperature
    def read(self):
        return DHT.read_retry(self.sensor, self.out)
    def pass_c(self, celsius):
        return celsius
    def pass_k(self, celsius):
        return celsius + 273.15
    def pass_f(self, celsius):
        return celsius * 9.0/5.0 + 32
```

ส่วนของเซนเซอร์วัดอุณหภูมิและความชื้น 2

```
import DHT22 as DHT22
import time
import pyrebase

out = 8
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
units = 'c'
sensor = DHT22.DHT22(out,units)
time.sleep(1)
running = True

while(running):
    try:
        humidity, temperature = sensor.get_value()
        if humidity is not None and temperature is not None:
            print ('Temp={0:0.1f}*C  Humidity={1:0.1f}%'.format(temperature, hu$
            time.sleep(2)
            temp = str('{:.1f}'.format(temperature))
            hum = str('{:.1f}'.format(humidity))
            print (temp)
            print (hum)
            time.sleep(1)
    except KeyboardInterrupt:
        running = False
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนของมอเตอร์

```
import RPi.GPIO as io
import time

io.setmode(io.BCM)

io.setup(23, io.OUT)

io.setup(24, io.OUT)

io.setup(21,io.IN)

io.setup(18, io.OUT)

p=io.PWM(18,1000)
p.start(100)

for i in range(0,1):
  io.output(23,1)
  io.output(24,0)
  #time.sleep(5)
  if io.input(21) == 1:
        io.output(23,0)
        io.output(24,0)
  io.output(23,0)
  io.output(24,1)
  if io.input(21) == 1:
        io.output(23,0)
        io.output(24,0)
  #time.sleep(5)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนของลิมิตสวิตช์

```
import RPi.GPIO as io
import time

io.setmode(io.BCM)

io.setup(16,io.IN)

io.setup(20,io.IN)

io.setup(21,io.IN)

while True:
 print(io.input(21)

 time.sleep(1)
```

การทำงานตามสภาพภูมิอากาศ

```
import smbus
import RPi.GPIO as io
import time
from gpiozero import InputDevice

norain = InputDevice(15)

io.setmode(io.BCM)

io.setup(23, io.OUT)

io.setup(24, io.OUT)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
io.setup(16, io.IN)

io.setup(20, io.IN)

io.setup(21, io.IN)

bus = smbus.SMBus(1)
bus.write_byte_data(0x39, 0x00 | 0x80, 0x03)
bus.write_byte_data(0x39, 0x01 | 0x80, 0x02)
time.sleep(0.5)

while True:

  data = bus.read_i2c_block_data(0x39, 0x0C | 0x80, 2)
  data1 = bus.read_i2c_block_data(0x39, 0x0E | 0x80, 2)

  ch0 = data[1] * 256 + data[0]
  ch1 = data1[1] * 256 + data1[0]

  if io.input(16) == 1:
    if norain.is_active==1  and ch1>0:
      io.output(23,1)
      io.output(24,0)

      if io.input(21) == 1:
          io.output(23,0)
          io.output(24,0)

    else:
      io.output(23,0)
      io.output(24,1)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    if io.input(20) == 1:
        io.output(23,0)
        io.output(24,0)

if io.input(16) != 1:
    io.output(23,0)
    io.output(24,1)

    if io.input(20) == 1:
        io.output(23,0)
        io.output(24,0)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข

ชุดคำสั่งที่ใช้ส่งข้อมูลเซนเซอร์จาก Raspberry Pi ไปยัง Firebase Realtime Database

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนของเซนเซอร์ตรวจวัดความเข้มแสง

```
import pyrebase
import smbus
import time

bus = smbus.SMBus(1)
bus.write_byte_data(0x39, 0x00 | 0x80, 0x03)
bus.write_byte_data(0x39, 0x01 | 0x80, 0x02)
time.sleep(0.5)

config = {
  "apiKey": "gOgPfhpArQc5LSI4rhQogrtUzk45kZCoaCU4Tksg",

  "authDomain": "testfire-e5c0f.firebaseapp.com",

  "databaseURL": "https://testfire-e5c0f-default-rtdb.firebaseio.com/",

  "storageBucket": "testfire-e5c0f.appspot.com"

}

firebase = pyrebase.initialize_app(config)
db = firebase.database()

print ("Send Data to Firebase Using Raspberry Pi")
print ("————————————————————-")
print ()

while True:

  data = bus.read_i2c_block_data(0x39, 0x0C | 0x80, 2)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  data1 = bus.read_i2c_block_data(0x39, 0x0E | 0x80, 2)

# Convert the data
  ch0 = data[1] * 256 + data[0]
  ch1 = data1[1] * 256 + data1[0]

  print ("Full Spectrum(IR + Visible) :%d lux" %ch0)
  lux = str(ch0)

  data = {
  "intensity": lux,
  }

  db.update(data)
  time.sleep(1)
```

ส่วนของโมดูลตรวจจับน้ำฝน

```
import time
from time import sleep
from gpiozero import InputDevice
import pyrebase

norain = InputDevice(15)

rain = []

config = {
  "apiKey": "gOgPfhpArQc5LSI4rhQogrtUzk45kZCoaCU4Tksg",
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  "authDomain": "testfire-e5c0f.firebaseapp.com",

  "databaseURL": "https://testfire-e5c0f-default-rtdb.firebaseio.com/",

  "storageBucket": "tes = []tfire-e5c0f.appspot.com"

}

firebase = pyrebase.initialize_app(config)
db = firebase.database()

print("Send Data to Firebase Using Raspberry Pi")
print("————————————————————")
print()

while True:
  if not norain.is_active:
    print ("it raining")
    rain0 = norain.is_active
    time.sleep(1)

  else:
    print ("it isn't raining")
    rain0 = norain.is_active
    time.sleep(1)

  rain = str(rain0)

  data = {
    "rainstatus": rain,
  }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  db.update(data)
  time.sleep(2)
```

ส่วนของเซนเซอร์วัดความชื้นและอุณหภูมิ

```
import DHT22 as DHT22
import time
import pyrebase

out = 8

units = 'c'

sensor = DHT22.DHT22(out,units)

time.sleep(1)

running = True

config = {
  "apiKey": "gOgPfhpArQc5LSI4rhQogrtUzk45kZCoaCU4Tksg",

  "authDomain": "testfire-e5c0f.firebaseapp.com",

  "databaseURL": "https://testfire-e5c0f-default-rtdb.firebaseio.com/",

  "storageBucket": "testfire-e5c0f.appspot.com"

}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
firebase = pyrebase.initialize_app(config)
db = firebase.database()

print ("Send Data to Firebase Using Raspberry Pi")
print ("————————————————————-")
print ()

while(running):

    try:

        humidity, temperature = sensor.get_value()

        if humidity is not None and temperature is not None:

            print ('Temp={0:0.1f}*C  Humidity={1:0.1f}%'.format(temperature, hu$

            time.sleep(2)

            temp = str('{:.1f}'.format(temperature))
            hum = str('{:.1f}'.format(humidity))
            print (temp)
            print (hum)

            data = {
              "temperature": temp,
              "humidity": hum,
            }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        db.update(data)
        time.sleep(1)

except KeyboardInterrupt:

    running = False
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ค

ชุดคำสั่งที่ใช้ควบคุมการทำงานของมอเตอร์ตามเงื่อนไขทั้งหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
import smbus
import RPi.GPIO as io
import time
import pyrebase
from gpiozero import InputDevice

config = {
  "apiKey": "gOgPfhpArQc5LSI4rhQogrtUzk45kZCoaCU4Tksg",

  "authDomain": "testfire-e5c0f.firebaseapp.com",

  "databaseURL": "https://testfire-e5c0f-default-rtdb.firebaseio.com/",

  "storageBucket": "testfire-e5c0f.appspot.com"
}

firebase = pyrebase.initialize_app(config)
db = firebase.database()

print ("Send Data to Firebase Using Raspberry Pi")
print ("————————————————————-")
print ()

#####################################################################

norain = InputDevice(15)

io.setmode(io.BCM)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
io.setup(23, io.OUT)

io.setup(24, io.OUT)

io.setup(16, io.IN)

io.setup(20, io.IN)

io.setup(21, io.IN)

bus = smbus.SMBus(1)

bus.write_byte_data(0x39, 0x00 | 0x80, 0x03)

bus.write_byte_data(0x39, 0x01 | 0x80, 0x02)

time.sleep(0.5)

while True:

  showwork = db.child("working").get()

  y = showwork.val()

  data = bus.read_i2c_block_data(0x39, 0x0C | 0x80, 2)

  data1 = bus.read_i2c_block_data(0x39, 0x0E | 0x80, 2)

  ch0 = data[1] * 256 + data[0]
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
ch1 = data1[1] * 256 + data1[0]

if y==("auto"):

  if io.input(16) == 1:
    if norain.is_active==1  and ch0>50:

      io.output(23,1)
      io.output(24,0)

      if io.input(21) == 1:

        io.output(23,0)
        io.output(24,0)

    else:

      io.output(23,0)
      io.output(24,1)

      if io.input(20) == 1:

        io.output(23,0)
        io.output(24,0)

  if io.input(16) != 1:

    io.output(23,0)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    io.output(24,1)

    if io.input(20) == 1:

      io.output(23,0)
      io.output(24,0)

if y == ("manual"):

  showmot = db.child("motor").get()

  x = showmot.val()

  if io.input(16) == 1:
   if x==("inside"):

     print ("inside")
     io.output(23,0)
     io.output(24,1)

     if io.input(20) == 1:

       io.output(23,0)
       io.output(24,0)

    else:

      print ("outside")
      io.output(23,1)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    io.output(24,0)

    if io.input(21) == 1:

      io.output(23,0)
      io.output(24,0)

if io.input(16) != 1:

  io.output(23,0)
  io.output(24,1)

  if io.input(20) == 1:

    io.output(23,0)
    io.output(24,0)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ง

ชุดคำสั่งที่ใช้เริ่มการทำงานอัตโนมัติเมื่อเปิดใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

sudo nano /etc/xdg/lxsession/LXDE-pi/autostart

@lxpanel --profile LXDE

@pcmanfm --desktop --profile LXDE

@lxterminal

@leafpad

@xscreensaver -no-splash@

@python3 /home/pi/firebase/testlight.py

@python3 /home/pi/firebase/testrain.py

@python3 /home/pi/firebase/main.py

@python3 /home/pi/firebase/finalproject.py

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก จ

ชุดคำสั่งเปลี่ยนหน้าบนแอปพลิเคชัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
package com.example.acdr;

import androidx.appcompat.app.AppCompatActivity;
import androidx.appcompat.app.AppCompatDelegate;

import android.content.Intent;
import android.content.pm.ActivityInfo;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {
    private static final int TIME_INTERVAL = 2000;
    private long mBackPressed;

    Button Display, Control;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        this.setRequestedOrientation(ActivityInfo.SCREEN_ORIENTATION_PORTRAIT);

AppCompatDelegate.setDefaultNightMode(AppCompatDelegate.MODE_NIGHT_NO);

        Display = findViewById(R.id.Display);
        Control = findViewById(R.id.Control);

        Display.setOnClickListener(new View.OnClickListener() {
            @Override
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            public void onClick(View view) {
                Intent intent1 = new Intent(getApplicationContext(), ACDR_Display.class);
                startActivity(intent1);
            }
        });
        Control.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Intent intent2 = new Intent(getApplicationContext(), ACDR_Control.class);
                startActivity(intent2);
            }
        });
    }

    @Override
    public void onBackPressed() {
        if (mBackPressed + TIME_INTERVAL > System.currentTimeMillis()) {
            super.onBackPressed();
            return;
        } else {
            Toast.makeText(getBaseContext(), "กดปุ่ม ย้อนกลับ อีกครั้ง เพื่อออกจากโปรแกรม",
Toast.LENGTH_LONG).show();
            mBackPressed = System.currentTimeMillis();
        }
    }
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ฉ

ชุดคำสั่งที่ใช้แสดงผลลัพธ์ของเซนเซอร์บนแอปพลิเคชัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
package com.example.acdr;

import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
import androidx.appcompat.app.AppCompatDelegate;

import android.content.pm.ActivityInfo;
import android.os.Bundle;
import android.widget.TextView;

import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.ValueEventListener;

public class ACDR_Display extends AppCompatActivity  {

   private TextView temp;
   private TextView intensity;
   private TextView humidity;
   private TextView rain;
   DatabaseReference dtemp;
   DatabaseReference dbright;
   DatabaseReference dhumidity;
   DatabaseReference drain;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_acdr_display);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        temp = (TextView) findViewById(R.id.temperature);
        intensity = (TextView) findViewById(R.id.brightness);
        humidity = (TextView) findViewById(R.id.humidity);
        rain = (TextView) findViewById(R.id.rain);
        this.setRequestedOrientation(ActivityInfo.SCREEN_ORIENTATION_PORTRAIT);

AppCompatDelegate.setDefaultNightMode(AppCompatDelegate.MODE_NIGHT_NO);

        dtemp = FirebaseDatabase.getInstance().getReference();
        dtemp.addValueEventListener(new ValueEventListener() {
            @Override
            public void onDataChange(@NonNull DataSnapshot snapshot) {
                String statustemp = snapshot.child("temperature").getValue().toString();
                temp.setText(statustemp);
            }

            @Override
            public void onCancelled(@NonNull DatabaseError error) {

            }
        });

        dbright = FirebaseDatabase.getInstance().getReference();
        dbright.addValueEventListener(new ValueEventListener() {
            @Override
            public void onDataChange(@NonNull DataSnapshot snapshot) {
                String statusbright = snapshot.child("intensity").getValue().toString();
                intensity.setText(statusbright);
            }

            @Override
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    public void onCancelled(@NonNull DatabaseError error) {

    }
});

dhumidity = FirebaseDatabase.getInstance().getReference();
dhumidity.addValueEventListener(new ValueEventListener() {
    @Override
    public void onDataChange(@NonNull DataSnapshot snapshot) {
        String statushumidity = snapshot.child("humidity").getValue().toString();
        humidity.setText(statushumidity);
    }

    @Override
    public void onCancelled(@NonNull DatabaseError error) {

    }
});

drain = FirebaseDatabase.getInstance().getReference();
drain.addValueEventListener(new ValueEventListener() {
    @Override
    public void onDataChange(@NonNull DataSnapshot snapshot) {
        String statusrain = snapshot.child("rainstatus").getValue().toString();
        rain.setText(statusrain);
    }

    @Override
    public void onCancelled(@NonNull DatabaseError error) {

    }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        });

    }
}
```


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ช

ชุดคำสั่งที่ใช้ควบคุมการทำงานของมอเตอร์บนแอปพลิเคชัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
public class ACDR_Control extends AppCompatActivity {

    SwitchCompat motor;

    ImageView imageView;

    SwitchCompat work;


    DatabaseReference dwork;

    DatabaseReference dmotor;


    BottomNavigationView bottomNavigationView;


    boolean check = false;


    private boolean workcheck = true;

    private boolean motorcheck = true;


    String stawork;

    String stamotor;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_acdr_control);


        work = findViewById(R.id.switch1);
        motor = findViewById(R.id.switch2);
        imageView = findViewById(R.id.imageView7);
        this.setRequestedOrientation(ActivityInfo.SCREEN_ORIENTATION_PORTRAIT);
AppCompatDelegate.setDefaultNightMode(AppCompatDelegate.MODE_NIGHT_NO
);


        work.setOnCheckedChangeListener(new
CompoundButton.OnCheckedChangeListener() {
            @Override
            public void onCheckedChanged(CompoundButton buttonViwe, boolean
isChecked) {


                if (isChecked){
                    FirebaseDatabase database = FirebaseDatabase.getInstance();
                    DatabaseReference statuswork = database.getReference("working");
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                statuswork.setValue("auto");
            }else {
                FirebaseDatabase database = FirebaseDatabase.getInstance();
                DatabaseReference statuswork = database.getReference("working");
                statuswork.setValue("manual");
            }

        }
    });

    dwork = FirebaseDatabase.getInstance().getReference();
    dwork.addValueEventListener(new ValueEventListener() {
        @Override
        public void onDataChange(@NonNull @NotNull DataSnapshot snapshot)
{
            stawork = snapshot.child("working").getValue(String.class);
            if(workcheck){
                if (stawork.equals("auto")){
                    work.setChecked(true);
                } else if (stawork.equals("manual")){
                    work.setChecked(false);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                }

                workcheck = false;

            }

        }


        @Override

        public void onCancelled(@NonNull DatabaseError error) {


        }

    });


    motor.setOnCheckedChangeListener(new
CompoundButton.OnCheckedChangeListener() {

        @Override

        public void onCheckedChanged(CompoundButton buttonViwe, boolean
isCheck) {


            if (isCheck) {

                FirebaseDatabase database = FirebaseDatabase.getInstance();

                DatabaseReference statusmotor = database.getReference("motor");

                statusmotor.setValue("outside");
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                } else {
                    FirebaseDatabase database = FirebaseDatabase.getInstance();
                    DatabaseReference statusmotor = database.getReference("motor");
                    statusmotor.setValue("inside");
                }

            }
        });

        dmotor = FirebaseDatabase.getInstance().getReference();
        dmotor.addValueEventListener(new ValueEventListener() {
            @Override
            public void onDataChange(@NonNull @NotNull DataSnapshot snapshot)
{
                stamotor = snapshot.child("motor").getValue(String.class);
                if(motorcheck){
                    if(stamotor.equals("outside")){
                        motor.setChecked(true);

imageView.setImageDrawable(getResources().getDrawable(R.drawable.outside));
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                } else if (stamotor.equals("inside")){

                    motor.setChecked(false);

imageView.setImageDrawable(getResources().getDrawable(R.drawable.inside));
                }

                workcheck = false;

            }


        }

        @Override

        public void onCancelled(@NonNull DatabaseError error) {


        }

    });



        }

    }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้