เครื่องเขย่าแบบควบคุมอุณหภูมิและแสงสว่าง
ILLUMINATING INCUBATOR SHAKER

ณัฐกิตติ์ ติวะวงศ์

ณัฐพงศ์ ถึงสวัสดิ์

ชูชีพ ขยรประดับกุล

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมการวัดคุม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2557

เครื่องเขย่าแบบควบคุมอุณหภูมิและแสงส่องสว่าง

ILLUMINATING INCUBATOR SHAKER

ณัฐกิตติ์ ติวะวงศ์
ณัฐพงศ์ อึ้งสวัสดิ์
สุกิจ ขจรประดับกุล

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมการวัดคุม
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2557

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ILLUMINATING INCUBATOR SHAKER

NUTTAKIT TIWAWONG

NUTTAPONG UENGSAWAT

SUKIT KAJONPRADAPKUL

A THESIS SUBMITTED IN PARTIAL FULFILLMENT

OF THE REQUIREMENT FOR THE DEGREE OF

BACHELOR OF ENGINEERING IN INSTRUMENTATION ENGINEERRING

FACULTY OF ENGINEERING

KING MONGKUT'S INSTITUE OF TECHNOLOGY LARDKRABANG

ACADEMIC YEAR 2014

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2557

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ใบรับรองปริญญานิพนธ์

.....

หัวข้อปริญญานิพนธ์	เครื่องเขย่าแบบควบคุมอุณหภูมิและแสงสว่าง		
	ILLUMINATING INCUBATOR SHAKER		
นักศึกษาผู้จัดทำ	นายณัฐกิตติ์	ติวะวงศ์	**รหัสนักศึกษา** 54010393
	นายณัฐพงศ์	อึ้งสวัสดิ์	**รหัสนักศึกษา** 54010421
	นายสุกิจ	ขจรประดับกุล	**รหัสนักศึกษา** 54011383
ปริญญา	วิศวกรรมศาสตรบัณฑิต		
สาขาวิชา	วิศวกรรมการวัดคุม		
ปีการศึกษา	2557		

อาจารย์ผู้ควบคุมปริญญานิพนธ์	**ลายมือชื่อ**
อาจารย์สุธรรม สัทธรรมสกุล	
อาจารย์นรินทร์ ธรรมรักษ์วัฒนะ	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	เครื่องเขย่าแบบควบคุมอุณหภูมิและแสงสว่าง ILLUMINATING INCUBATOR SHAKER		
นักศึกษาผู้จัดทำ	นายณัฐกิตติ์	ติวะวงศ์	**รหัสนักศึกษา** 54010393
	นายณัฐพงศ์	อึ้งสวัสดิ์	**รหัสนักศึกษา** 54010421
	นายสุกิจ	ขจรประดับกุล	**รหัสนักศึกษา** 54011383
อาจารย์ที่ปรึกษา	อาจารย์สุธรรม	สัทธรรมสกุล	
	อาจารย์นรินทร์	ธรรมรักษ์วัฒนะ	
ปีการศึกษา	2557		

บทคัดย่อ

โครงการนี้นำเสนอ เครื่องเขย่าแบบควบคุมอุณหภูมิและแสงส่องสว่าง สำหรับใช้ในงานวิจัยเพาะเลี้ยงสาหร่าย เพื่อศึกษาสภาวะที่เหมาะสมต่อการเจริญเติบโต และความสามารถในการผลิตน้ำมัน โดยเครื่องเขย่านี้สามารถควบคุมตัวแปรต่าง ๆ ที่มีผลต่อการเจริญเติบโตของสาหร่ายได้ โดยใช้บอร์ดอาร์ดูโน่เป็นหัวใจหลักใน การควบคุมอุณหภูมิ และ ความส่องสว่างภายในตัวเครื่อง โดยสามารถแสดงผล และกำหนดค่าควบคุมทั้งอุณหภูมิและความสว่างได้ผ่านทางหน้าจอระบบสัมผัส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Thesis Title	ILLUMINATING INCUBATOR SHAKER	
Authors	Mr. Nuttakit	Tiwawong
	Mr. Nuttapong	Uengsawat
	Mr. Sukit	Kajonpradapkul
Thesis Advisor	Mr. Sutham	Satthamsakul
	Mr.Narin	thammarakwattana
Year	2014	

ABSTRACT

This project presents an illuminating Incubator Shaker that capable of control temperature and intensity of illumination. The illuminating incubator shaker use Arduino board for control temperature and illumination. The illuminating Incubator shaker can be controlled and displayed by 4D touch screen.

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้ สำเร็จลุล่วงไปด้วยดี เพราะอาจารย์สุธรรม สัทธรรมสกุล อาจารย์ที่ปรึกษาโครงงาน ที่ได้ให้ความเมตตา และคำแนะนำแก่ผู้จัดทำมาโดยตลอด อีกทั้งยังเอื้อเฟื้ออุปกรณ์และเครื่องมือต่าง ๆ ในการทำปริญญานิพนธ์นี้ ผู้จัดทำรู้สึกซาบซึ้งและกราบขอบพระคุณเป็นอย่างสูง

ขอขอบพระคุณอาจารย์และเจ้าหน้าที่ภาควิชาวิศวกรรมการวัดคุมทุกท่าน ที่ได้ให้คำแนะนำและช่วยเหลืออันเป็นประโยชน์ต่อการทำปริญญานิพนธ์ฉบับนี้ ที่ลืมเสียมิได้ขอกราบขอบพระคุณคุณพ่อและคุณแม่อันเป็นที่รักและเคารพอย่างยิ่งที่สนับสนุนเป็นแรงบันดาลใจ ในการทำปริญญานิพนธ์ฉบับนี้

คุณค่าและประโยชน์อันพึงมีจากปริญญานิพนธ์ฉบับนี้ ขอมอบให้แด่ผู้มีพระคุณทุกท่าน ที่ให้ความช่วยเหลือและเป็นกำลังใจให้ จนปริญญานิพนธ์ฉบับนี้เสร็จสมบูรณ์ และมอบให้แก่ผู้สนใจการศึกษาทั้งมวล

คณะผู้จัดทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อภาษาไทย I

บทคัดย่อภาษาอังกฤษ II

กิตติกรรมประกาศ III

สารบัญ IV

สารบัญรูป VIII

บทที่ 1 บทนำ 1

1.1 ความสำคัญของปริญญานิพนธ์ 1

1.2 วัตถุประสงค์ของปริญญานิพนธ์ 2

1.3 ขอบเขตของปริญญานิพนธ์ 2

1.4 ขั้นตอนการดำเนินการ 2

1.5 ประโยชน์ที่คาดว่าจะได้รับ 3

บทที่ 2 ทฤษฎีและหลักการที่เกี่ยวข้อง 4

2.1 บทนำ 4

2.2 บอร์ดอาร์ดูโน่เมก้า 4

2.2.1 การจ่ายพลังงาน 5

2.2.2 หน่วยความจำ 5

2.2.3 อินพุตและเอาท์พุต 6

2.2.4 การสื่อสาร 6

2.3 การสื่อสารด้วยพอร์ต UART 7

2.4 การสื่อสารข้อมูลด้วย I^2C 8

2.4.1 คุณสมบัติโดยทั่วไปของบัส I^2C 8

2.4.2 หลักการของบัส I^2C 8

2.4.3 สภาวะที่เกิดขึ้นบนบัส I^2C 9

2.4.4 การทำงานบนบัส I^2C 10

2.4.4.1 การอ้างอิงถึงแบบ 7 บิต (7-bit addressing) 10

2.4.4.2 การเข้าถึงแบบ 10 บิต (10-bit addressing) 11

2.5 การสื่อสารข้อมูลด้วย SPI 11

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

2.6 การสื่อสารด้วยสายสัญญาณ 1 เส้น....12
2.6.1 การต่อใช้งานระหว่างอุปกรณ์มาสเตอร์และสเลฟ....13
2.6.1.1 การเขียนข้อมูลไปยังอุปกรณ์สเลฟ....13
2.6.1.2 การอ่านข้อมูลจากอุปกรณ์สเลฟ....14
2.7 หน้าจอระบบสัมผัส....15
2.7.1 โปรแกรมสำหรับออกแบบหน้าจอระบบสัมผัส....15
2.8 อุปกรณ์บันทึกข้อมูล....16
2.8.1 คุณสมบัติของอุปกรณ์บันทึกข้อมูล....17
2.8.2 ส่วนประกอบหลักภายในอุปกรณ์บันทึกข้อมูล....17
2.9 เซ็นเซอร์วัดอุณหภูมิ....18
2.10 เซ็นเซอร์วัดความสว่าง....19
2.10.1 คุณสมบัติของ BH 1750 FVI....19
2.11 หลอดฟลูออเรสเซนต์....20
2.12 บัลลาสต์อิเล็กทรอนิกส์....21
2.13 ออปแอมป์....21
2.14 ไอซี 555....24
2.14.1 การจัดวงจรใช้งานไอซี 555....24

บทที่ 3 การออกแบบ....25
3.1 การออกแบบและการทำงานโดยรวมของระบบ....25
3.2 การออกแบบแผนผังการทำงานของระบบ....26
3.2.1 โครงสร้างการทำงานหลักของระบบ....26
3.2.2 โครงสร้างการทำงานของฟังก์ชันการทำงานตามสถานะ....27
3.2.3 โครงสร้างการทำงานของฟังก์ชันย่อยระบบควบคุมแสงส่องสว่าง....30
3.2.4 โครงสร้างการทำงานของฟังก์ชันย่อยของระบบควบคุมอุณหภูมิ....31
3.2.5 โครงสร้างการทำงานของฟังก์ชันย่อยการตั้งเวลา....34
3.2.6 โครงสร้างการทำงานของฟังก์ชันย่อยของส่วนบันทึกข้อมูล....36
3.2.7 โครงสร้างการทำงานของฟังก์ชันย่อยของฟังก์ชันกลางวันและกลางคืน....39

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

3.3 การออกแบบหน้าจอสำหรับใช้งาน....41
3.3.1 โครงสร้างการเชื่อมโยงของหน้าจอระบบสัมผัส....41
3.3.2 หน้าจอเริ่มต้น....43
3.3.3 หน้าจอแสดงผล....44
3.3.4 หน้าจอตั้งค่าแสงสว่าง....44
3.3.5 หน้าจอตั้งค่าอุณหภูมิ....45
3.3.6 หน้าจอตั้งค่าเวลา....45
3.3.7 หน้าจอแสดงสถานะขณะกำลังเครื่องทำงาน....46

บทที่ 4 การดำเนินงาน....47
4.1 การดำเนินงานและการเชื่อมต่ออุปกรณ์....47
4.2 ระบบควบคุมแสงสว่าง....47
4.2.1 โครงสร้างระบบควบคุมแสงสว่าง....47
4.2.2 การรับค่าความสว่างจากเซ็นเซอร์วัดแสง (BH1750FVI)....48
4.2.3 การใช้งานบัลลาสต์อิเล็กทรอนิกส์กับหลอดฟลูออเรสเซนต์....49
4.2.4 วงจรแปลงแรงดัน....49
4.3 ระบบควบคุมอุณหภูมิ....50
4.3.1 โครงสร้างของระบบควบคุมอุณหภูมิ....50
4.3.2 การรับค่าอุณหภูมิจากเซ็นเซอร์อุณหภูมิ (DS18B20)....51
4.3.3 การเพิ่มอุณหภูมิภายในระบบด้วยฮีทเตอร์และพัดลม....52
4.4 การตั้งเวลาใช้งาน....52
4.4.1 การดัดแปลงชิพค่าเวลานาฬิกาจริง....52
เพื่อใช้งานกับบอร์ดอาร์ดูโน่เมก้า
4.4.2 โครงสร้างการแสดงผลและตั้งเวลาผ่านหน้าจอระบบสัมผัส....53
4.5 การบันทึกข้อมูลด้วยอุปกรณ์บันทึกข้อมูล....53
4.5.1 การดัดแปลงส่วนของการบันทึกข้อมูล....54
ให้สามารถนำมาใช้งานร่วมกับบอร์ดอาร์ดูโน่เมก้า
4.5.2 โครงสร้างการบันทึกข้อมูลด้วยอุปกรณ์บันทึกข้อมูล....54

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

บทที่ 5 ผลการทดลอง **55**

5.1 บทนำ 55

5.2 การทดลองและผลการทดลองการทำงานของระบบควบคุมอุณหภูมิ 55

5.2.1 การทดลองการทำงานของระบบควบคุมอุณหภูมิ 55

5.2.2 ผลการทดลองการทำงานของระบบควบคุมอุณหภูมิ 55

5.3 การทดลองและผลการทดลองการทำงานของระบบควบคุมแสงสว่าง 56

5.3.1 การทดลองการทำงานของระบบควบคุมแสงสว่าง 56

5.3.2 ผลการทดลองการทำงานของระบบควบคุมแสงสว่าง 56

5.4 การทดลองและผลการทดลองระบบควบคุม 57
แสงสว่างในโหมดกลางวันกลางคืน

5.4.1 การทดลองระบบควบคุมแสงสว่างในโหมดกลางวันกลางคืน 57

5.4.2 ผลการทดลองระบบควบคุมแสงสว่างในโหมดกลางวันกลางคืน 58

5.5 การทดลองและผลการทดลองการทำงาน 59
ของระบบควบคุมแสงสว่างร่วมกับอุณหภูมิ

5.5.1 การทดลองหาการทดลองการทำงานของระบบ 59
ควบคุมแสงสว่างร่วมกับอุณหภูมิ

5.5.2 ผลการทดลองการทำงานของระบบ 59
ควบคุมแสงสว่างร่วมกับอุณหภูมิ

บทที่ 6 สรุปผลการทดลองและข้อเสนอแนะ **62**

6.1 สรุปผล 62

6.2 ปัญหาและแนวทางแก้ไข 62

6.3 ข้อเสนอแนะ 63

บรรณานุกรม **64**

ภาคผนวก **66**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่ **หน้า**

2.1 บอร์ดอาร์ดูโน่เมก้า....4
2.2 รูปแบบการสื่อสารของ UART....7
2.3 การเชื่อมต่อการสื่อสารแบบ SPI ระหว่างอุปกรณ์ มาสเตอร์-สเลฟ....11
2.4 การเชื่อมต่อการสื่อสารแบบ SPI ระหว่างอุปกรณ์มาสเตอร์-สเลฟหลายตัว....12
2.5 การใช้บัสสายสัญญาณ 1 เส้นต่อร่วมกับอุปกรณ์มาสเตอร์-สเลฟหลายตัว....13
2.6 การติดต่อสื่อสารแบบสายสัญญาณ 1 เส้นกับอุปกรณ์สเลฟ....13
2.7 การเขียนข้อมูลไปยังอุปกรณ์สเลฟ....14
2.8 การอ่านข้อมูลจากบัสสัญญาณ 1 เส้น....14
2.9 การเชื่อมต่อระหว่างจอสัมผัสกับโฮสต์....16
2.10 ส่วนประกอบภายในอุปกรณ์บันทึกข้อมูล....17
2.11 โครงสร้างและขาของ DS18B20....18
2.12 วงจรภายในของ BH 1750 FVI....19
2.13 ภายในของหลอดฟลูออเรสเซนต์....20
2.14 การต่อหลอดฟลูออเรสเซนต์เข้ากับบัลลาสต์และสตาร์ทเตอร์....20
2.15 การต่อหลอดฟลูออเรสเซนต์เข้ากับบัลลาสต์อิเล็กทรอนิกส์ แบบ 2 หลอด....21
2.16 ออปแอมป์เบอร์ LM741....22
2.17 สัญลักษณ์ของออปแอมป์....22
2.18 วงจรภายในออปแอมป์....22
2.19 วงจรสมมูลอย่างง่ายของออปแอมป์....23
2.20 วงจรสมมูลอย่างง่ายของออปแอมป์ในอุดมคติ....23
2.21 ไอซี 555....24
3.1 บล๊อกไดอะแกรมโครงสร้างการทำงานของระบบ....25
3.2 โครงสร้างการทำงานหลักของระบบ....26
3.3 โครงสร้างการทำงานของฟังก์ชันการทำงานตามสถานะ....29
3.4 โครงสร้างการทำงานของฟังก์ชันย่อยระบบควบคุมแสงส่องสว่าง....30
3.5 โครงสร้างการทำงานของฟังก์ชันย่อยระบบควบคุมอุณหภูมิ....33
3.6 โครงสร้างการทำงานของฟังก์ชันย่อยการตั้งเวลา....35
3.7 โครงสร้างการทำงานของฟังก์ชันย่อยส่วนบันทึกข้อมูล....38
3.8 โครงสร้างการทำงานของฟังก์ชันย่อยกลางวันและกลางคืน....40

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่	หน้า
3.9 การเชื่อมโยงของหน้าจอระบบสัมผัส	42
3.10 หน้าจอเริ่มต้น	43
3.11 หน้าเตือนเมื่อฝาถูกเปิดอยู่	43
3.12 หน้าจอแสดงผล	44
3.13 หน้าจอตั้งค่าแสงสว่าง	44
3.14 หน้าจอตั้งค่าอุณหภูมิ	45
3.15 หน้าจอตั้งค่าเวลา	46
3.16 หน้าจอแสดงสถานะขณะเครื่องกำลังทำงาน	46
4.1 โครงสร้างของระบบควบคุมแสงสว่าง	48
4.2 การเชื่อมต่อ BH1750FVI เข้ากับบอร์ดอาร์ดูโน่	49
4.3 การเชื่อมต่อบัลลาสต์อิเล็กทรอนิกส์กับหลอดฟลูออเรสเซนต์	49
4.4 การเชื่อมต่อวงจรแปลงแรงดัน	50
4.5 โครงสร้างของระบบควบคุมอุณหภูมิ	51
4.6 การเชื่อมต่อเซ็นเซอร์อุณหภูมิเข้ากับบอร์ดอาร์ดูโน่	51
4.7 การดัดแปลงใช้งานชิพค่าเวลานาฬิกาจริง ให้สามารถทำงานร่วมกับบอร์ดอาร์ดูโน่เมก้า	52
4.8 การแสดงผลและตั้งเวลาผ่านหน้าจอระบบสัมผัส	53
4.9 การตั้งค่าขาที่ใช้ในการสื่อสารแบบ SPI ระหว่างส่วนบันทึกข้อมูลกับบอร์ดอาร์ดูโน่เมก้า	54
4.10 โครงสร้างการบันทึกข้อมูลด้วยอุปกรณ์บันทึกข้อมูล	54
5.1 ผลการทดลองการทำงานของระบบควบคุมอุณหภูมิ	56
5.2 ผลการทดลองการทำงานของระบบแสงสว่าง	57
5.3 ผลการทดลองระบบควบคุมแสงสว่างในโหมดกลางวันกลางคืน	59
5.4 ผลการทดลองการทำงานของระบบควบคุมแสงสว่าง ร่วมกับอุณหภูมิที่ 39 องศาเซลเซียส	60
5.5 ผลการทดลองการทำงานของระบบควบคุมแสงสว่าง ร่วมกับอุณหภูมิที่ 42 องศาเซลเซียส	60
5.6 ผลการทดลองการทำงานของระบบควบคุมแสงสว่าง ร่วมกับอุณหภูมิที่ 45 องศาเซลเซียส	61

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความสำคัญของปริญญานิพนธ์

ในยุคปัจจุบัน วิทยาการต่าง ๆ ก้าวหน้าไปอย่างรวดเร็ว เนื่องจากการพัฒนาด้านวิทยาศาสตร์และเทคโนโลยีในหลากหลายแขนง เช่น ชีววิทยา เคมี ฟิสิกส์ เป็นต้น ซึ่งแต่ละแขนงนั้นจะมีแนวทางในการพัฒนาที่เป็นเอกลักษณ์เฉพาะตัว และบางครั้งอาจมีการนำองค์ความรู้ในศาสตร์แขนงอื่นมาช่วยในการพัฒนาร่วมไปด้วย ส่งผลให้เกิดการพัฒนาองค์ความรู้ที่เป็นไปอย่างก้าวกระโดด โดยวิทยาการทางด้านชีววิทยานั้น มีการพัฒนามาอย่างต่อเนื่อง การค้นพบสิ่งมีชีวิตสายพันธุ์ใหม่ ๆ รวมถึงมีการพัฒนาทางด้านพันธุกรรมก็มีอยู่อย่างต่อเนื่อง ประกอบกับปัจจุบันเรื่องของพลังงานทดแทนก็เป็นงานวิจัยที่กำลังได้รับความนิยมเป็นอย่างมาก เนื่องจากมีการประเมินว่าจะเกิดวิกฤตการขาดแคลนพลังงานอย่างมากในอนาคต ดังนั้นจึงมีนักวิจัยทางด้านชีววิทยาส่วนหนึ่งจึงมีความสนใจเกี่ยวกับการวิจัยเรื่องของการผลิตไฮโดรเจนจากสาหร่าย และมีความจำเป็นที่จะต้องใช้เครื่องมือเข้ามาช่วยในการวิจัยหรือค้นคว้า เพราะสาหร่ายเป็นพืชน้ำที่ไม่สามารถเลี้ยงในน้ำนิ่งได้ ดังนั้นในการเลี้ยงจึงต้องมีการเขย่าให้น้ำมีการเคลื่อนไหวเพื่อให้สารอาหารในน้ำไม่เกิดการตกตะกอน รวมไปถึงการเพาะเลี้ยงก็ต้องมีการควบคุมแสงและอุณหภูมิที่เหมาะสม เครื่องเขย่าแบบควบคุมอุณหภูมิและแสงส่องสว่างถือเป็นเครื่องมือสำคัญชิ้นหนึ่ง ที่เข้ามามีส่วนช่วยในการวิจัยหรือการทดลองนี้ แต่เนื่องจากเครื่องมือดังกล่าวที่มีการนำเข้าจากต่างประเทศมีราคาที่สูงมาก จึงเป็นที่มาของโครงงานเครื่องเขย่าแบบควบคุมอุณหภูมิและแสงส่องสว่างที่จะนำเสนอในปริญญานิพนธ์เล่มนี้ โดยตัวเครื่องจะมีความสามารถในการควบคุมและจำลองสภาพแวดล้อม ภายในเครื่องได้ทั้งอุณหภูมิและ ความสว่างของแสง เพื่อให้ได้สภาพแวดล้อมที่ใกล้เคียงกับค่าที่ตั้งไว้มากที่สุด ทั้งนี้ ยังมีความสามารถในการเก็บบันทึกผลการเปลี่ยนแปลงที่เกิดขึ้นภายในระบบทั้งหมดไว้ในการ์ดหน่วยความจำ เพื่อประโยชน์ในการนำไปศึกษาและวิจัยต่อไปในอนาคต โดยมีบอร์ดอาร์ดูโน่เป็นตัวประมวลผลและควบคุม รวมถึงมีหน้าจอแบบสัมผัสเป็นตัวแสดงผลและสั่งควบคุมการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.2 วัตถุประสงค์ของปริญญานิพนธ์

1. เพื่อศึกษาการใช้งาน และเขียนโปรแกรมควบคุมอุปกรณ์ต่อพ่วงต่าง ๆ กับบอร์ดอาร์ดูโน่
2. เพื่อศึกษาการใช้งาน และเขียนโปรแกรมควบคุมการทำงานของหน้าจอแบบสัมผัสร่วมกับบอร์ดอาร์ดูโน่
3. ศึกษาการทำงานของเซนเซอร์ตรวจวัดความเข้มของแสง รวมถึงการเชื่อมต่อและเขียนโปรแกรม
4. ศึกษาการทำงานของเซนเซอร์ตรวจวัดอุณหภูมิ รวมถึงการเชื่อมต่อและเขียนโปรแกรม

1.3 ขอบเขตของปริญญานิพนธ์

1. ตัวเครื่องสามารถควบคุมการเปิดปิดการทำงานของมอเตอร์สำหรับเขย่าได้
2. ตัวเครื่องสามารถตรวจวัดอุณหภูมิภายในตัวเครื่อง รวมถึงสามารถควบคุม และรักษาระดับของอุณหภูมิให้อยู่ช่วงที่ใกล้เคียงกับอุณหภูมิที่ตั้งไว้ได้
3. ตัวเครื่องสามารถตรวจวัดปริมาณความเข้มของแสงภายในตัวเครื่อง รวมถึงสามารถควบคุมและรักษาระดับความเข้มของแสงให้อยู่ในช่วงใกล้เคียงกับความเข้มแสงที่ตั้งไว้ อีกทั้งยังสามารถจำลองการทำงานในลักษณะกลางวันและกลางคืน (day/night mode) ได้
4. ตัวเครื่องสามารถตั้งค่า วันที่ และเวลาปัจจุบันได้
5. ตัวเครื่องสามารถเก็บบันทึกค่าเหตุการณ์ต่าง ๆ (log) ที่เกิดการเปลี่ยนแปลงในขณะที่เครื่องกำลังทำงาน โดยบันทึกทั้งความเข้มแสง อุณหภูมิ และวันเวลาที่เกิดการเปลี่ยนแปลงนั้น
6. ตัวเครื่องสามารถแสดงผลทั้งอุณหภูมิและความเข้มของแสงที่ตรวจวัดได้และค่าที่ตั้งไว้ในขณะนั้น รวมถึงสามารถควบคุมและสั่งงานได้ผ่านทางหน้าจอระบบสัมผัส

1.4 ขั้นตอนการดำเนินการ

1. ทำการศึกษาทฤษฎีและหลักการที่เกี่ยวข้องกับโครงงานในเบื้องต้น
2. ศึกษาและค้นคว้าหาข้อมูลเกี่ยวกับอุปกรณ์ต่าง ๆ ที่จะนำมาใช้ในการออกแบบเครื่องเขย่าแบบควบคุมอุณหภูมิและแสงส่องสว่าง
3. วางแผนการและออกแบบรายละเอียดในส่วนต่าง ๆ ของตัวเครื่อง
4. ศึกษาการใช้งานและเขียนโปรแกรมสำหรับควบคุมบอร์ดอาร์ดูโน่
5. ศึกษาการใช้งานและการเขียนโปรแกรมบนหน้าจอระบบสัมผัส
6. เขียนโปรแกรมสำหรับควบคุมค่าความเข้มแสงสว่างและอุณหภูมิ
7. ออกแบบหน้าจอระบบสัมผัสสำหรับใช้ในการควบคุม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8. เขียนโปรแกรมสำหรับควบคุมค่าความเข้มแสงสว่างและอุณหภูมิ
9. ทำการทดลองเพื่อเก็บค่าเพื่อวิเคราะห์หาค่าที่เหมาะสมสำหรับการควบคุม
10. สรุปผลการทดลองและประเมินผล

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1. ได้ความรู้และประสบการณ์จากการใช้งาน และการเขียนโปรแกรมควบคุม รวมถึงฟังก์ชันต่าง ๆ ของอาร์ดูโน่
2. ได้ความรู้และประสบการณ์จากการใช้งาน และเขียนโปรแกรมควบคุมการทำงานหน้าจอแบบสัมผัสร่วมกับอาร์ดูโน่
3. ได้ความรู้และประสบการณ์จากการเขียนโปรแกรม เพื่อเชื่อมต่อกับอุปกรณ์ตรวจวัดอุณหภูมิและความเข้มแสง
4. ได้ความรู้และประสบการณ์จากการเขียนโปรแกรม ในการสั่งงานอุปกรณ์เพื่อควบคุมอุณหภูมิและค่าความเข้มของแสง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีและหลักการที่เกี่ยวข้อง

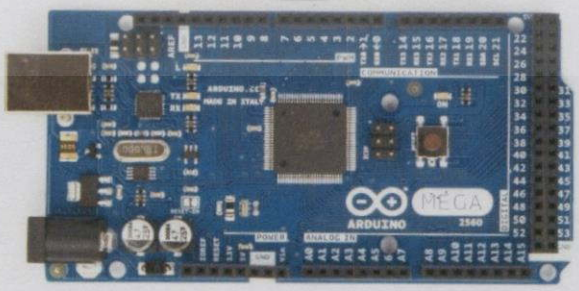
2.1 บทนำ

ในบทนี้จะกล่าวถึงทฤษฎีและหลักการ ที่เกี่ยวข้องสำหรับใช้ในการออกแบบและดำเนินการโครงงานเครื่องเขย่าแบบควบคุมอุณหภูมิและแสงส่องสว่าง ซึ่งประกอบด้วย บอร์ดอาร์ดูโน่ มาตรฐานการสื่อสารชนิดต่าง ๆ หน้าจอระบบสัมผัส ส่วนบันทึกข้อมูล เซ็นเซอร์วัดอุณหภูมิ เซ็นเซอร์วัดแสงสว่าง บัลลาสต์อิเล็กทรอนิกส์ และวงจรแปลงแรงดัน

2.2 บอร์ดอาร์ดูโน่ (Arduino Mega 2560)

อาร์ดูโน่เมก้า เป็นบอร์ดที่ใช้ไมโครคอนโทรลเลอร์ ATmega2560 เป็นส่วนควบคุม โดยจะมีดิจิตอลอินพุตและดิจิตอลเอาท์พุต รวมทั้งหมด 54 ขา (ซึ่งใน 54 ขา นั้นมีอยู่ 15 ขา ที่สามารถใช้เป็นสัญญาณเอาท์พุตแบบ PWM ได้) ขาแอนะล็อกอินพุต 16 ขา UARTs 4 ขา คริสตัลความถี่ 16 เมกกะเฮิรตซ์ (MHz) พอร์ตสำหรับเชื่อมต่อ USB แจ็คเชื่อมต่อแหล่งจ่ายไฟ หัวต่อแบบ ICSP และปุ่มรีเซ็ต โดยสามารถทำงานได้เมื่อมีการเชื่อมต่อกับคอมพิวเตอร์ด้วยสาย USB หรือ จ่ายไฟเป็นไฟฟ้ากระแสตรง ตัวบอร์ดอาร์ดูโน่เมก้าแสดงดังรูปที่ 2.1

บอร์ดอาร์ดูโน่เมก้านั้น จะแตกต่างจากบอร์ดรุ่นที่มีการผลิตก่อนหน้า เนื่องจากไม่ต้องใช้ชิพไดรเวอร์ของบริษัท FTDI เพื่อแปลง USB ไปเป็นพอร์ตอนุกรมสำหรับใช้ในการสื่อสาร จึงจำเป็นต้องออกแบบโปรแกรมควบคุมการทำงานของขาต่าง ๆ ขึ้นเอง ผ่านชิพ Atmega16U2 ที่ถูกโปรแกรมให้ทำหน้าแปลง USB ไปเป็นพอร์ตอนุกรมแทน โดย ATmega2560 จะติดต่อกับคอมพิวเตอร์ผ่าน Atmega16U2



รูปที่ 2.1 บอร์ดอาร์ดูโน่เมก้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.1 การจ่ายพลังงาน

บอร์ดอาร์ดูโน่เมก้านั้นรับการจ่ายพลังงานจากสาย USB หรือจากแหล่งพลังงานภายนอก โดยแหล่งจ่ายพลังงานนั้นจะถูกเลือกโดยอัตโนมัติ ซึ่งแหล่งจ่ายภายนอก (ที่ไม่ใช่ USB) นั้นจะเชื่อมต่อผ่านขั้วต่อแบบแจ็คขนาด 2.1 มิลลิเมตร เข้ากับพอร์ตเชื่อมต่อของบอร์ด และยังสามารถที่จะต่อแบตเตอรี่เข้าที่ขา Gnd และ V_{in} ของขั้วต่อสำหรับจ่ายไฟก็ได้ สำหรับแรงดันสูงสุดที่สามารถรองรับได้นั้นจะอยู่ในช่วง 6–20 โวลต์ แต่อย่างไรก็ตาม หากแหล่งจ่ายมีแรงดันต่ำกว่า 7 โวลต์ อาจส่งผลต่อแรงดันที่ขาจ่ายไฟ 5 โวลต์ อาจจ่ายไฟออกได้ไม่ถึง 5 โวลต์ และทำให้การทำงานของบอร์ดไม่เสถียร และหากมีแรงดันเกินกว่า 12 โวลต์แล้ว โวลต์เตจเรกูเลเตอร์ก็จะร้อนมากเกินไปและอาจส่งผลทำให้เกิดความเสียหายขึ้นที่บอร์ดได้ ดังนั้นช่วงแรงดันที่เหมาะสมกับการใช้งานมากที่สุดจะอยู่ในช่วง 7-12 โวลต์ โดยขาที่เกี่ยวกับส่วนของการจ่ายไฟจะมีดังต่อไปนี้

1. V_{in} แรงดันขาเข้าไปยังบอร์ดอาร์ดูโน่เมก้า เมื่อใช้แหล่งจ่ายภายนอก (ซึ่งแตกต่างจากแรงดัน 5 โวลต์ จาก USB หรือแหล่งจ่ายอื่น ๆ) โดยสามารถที่จะจ่ายแรงดันภายนอกเข้าที่ขานี้
2. 5V ขานี้จะจ่ายแรงดันเอาท์พุตที่ 5 โวลต์ ซึ่งได้จากเรกูเลเตอร์ของบอร์ด โดยหากจ่ายไฟโดยตรงเข้าขา 5V และ 3.3V อาจจะทำให้เกิดอันตรายแก่บอร์ดได้ จึงไม่แนะนำให้ทำ
3. 3.3V ขานี้จะจ่ายแรงดันเอาท์พุตที่ 3.3 โวลต์ ซึ่งได้จากเรกูเลเตอร์ของบอร์ด สามารถจ่ายกระแสได้สูงสุดที่ 50 มิลลิแอมป์ (mA)
4. GND ขากราวด์
5. IOREF ขานี้จะจ่ายแรงดันอ้างอิง โดยชิลด์ที่มาเชื่อมต่อกับบอร์ดจะสามารถเลือกใช้งานแรงดันที่เหมาะสมกับชิลด์ ที่ระดับแรงดัน 5 โวลต์ หรือ 3.3 โวลต์

2.2.2 หน่วยความจำ

ATmega2560 นั้นมีหน่วยความจำ 256 กิโลไบต์ (KB) ซึ่งจะใช้ 8 กิโลไบต์ (KB) เก็บโปรแกรมที่ทำหน้าที่ถ่ายโอนข้อมูลโปรแกรมสำหรับควบคุม (Arduino bootloader) และ ใช้สำหรับ SRAM ขนาด 8 กิโลไบต์ รวมถึงใช้กับ EEPROM ขนาด 4 กิโลไบต์ และส่วนที่เหลือจะใช้เก็บข้อมูลโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.3 อินพุตและเอาท์พุต

แต่ละขาของดิจิตอลเอาท์พุต 54 ขานั้นจะเลือกใช้เป็นอินพุตหรือเอาท์พุตก็ได้ โดยทำงานที่ระดับแรงดัน 5 โวลต์ ในแต่ละขาสามารถจ่ายหรือรับกระแสได้มากที่สุด 40 มิลลิแอมป์ (mA) และจะมีความต้านทานพูลอัพภายใน 20-50 กิโลโอห์ม โดยขาบางขาจะมีหน้าที่พิเศษ คือ

1. Serial0 : ขา 0 (RX) และ 1 (TX), Serial1 : ขา 19 (RX) และ 18 (TX), Serial2 : ขา 17 (RX) และ 16 (TX) และ Serial3 : ขา 15 (RX) และ 14 (TX) นั้นใช้เพื่อรับ (RX) และส่ง (TX) ข้อมูลแบบอนุกรมในลักษณะสัญญาณ TTL โดยขา 0 และ 1 จะเชื่อมต่อกับ ATmega16U2 ซึ่งทำหน้าที่แปลง USB เป็นพอร์ตอนุกรม
2. External interrupts : ขา 2, 3, 18, 19, 20 และ 21 โดยขาเหล่านี้สามารถที่จะขัดจังหวะ(interrupt) ได้ที่สถานะต่ำ(LOW)
3. PWM : ขา 2-13 และ ขา 44-46 ให้ค่าเอาท์พุทเป็นสัญญาณ PWM แบบ 8 บิต
4. SPI : 50(SS), 51(MOSI), 52(SCK) และ 53(SS) ขาเหล่านี้จะสนับสนุนการสื่อสารแบบ SPI
5. LED : ขา 13 เป็นขาที่เชื่อมต่ออยู่กับไดโอดเปล่งแสง (LED) เมื่อค่าที่ขามีสถานะสูง (High) ไฟจะติด เมื่อค่าที่ขามีสถานะต่ำ (Low) ไฟจะดับ
6. TWI : ขา 20 (SDA) และขา 21 (SCL) จะสนับสนุนการสื่อสารแบบ I^2C หรือ TWI
7. AREF ค่าแรงดันอ้างอิงสำหรับอินพุตแบบแอนะล็อก
8. Reset นำค่าสถานะต่ำเข้าที่ขานี้จะเป็นการรีเซ็ตไมโครคอนโทรลเลอร์ โดยปกติจะเพิ่มปุ่มเข้าไป เพื่อให้ใช้งานได้สะดวก

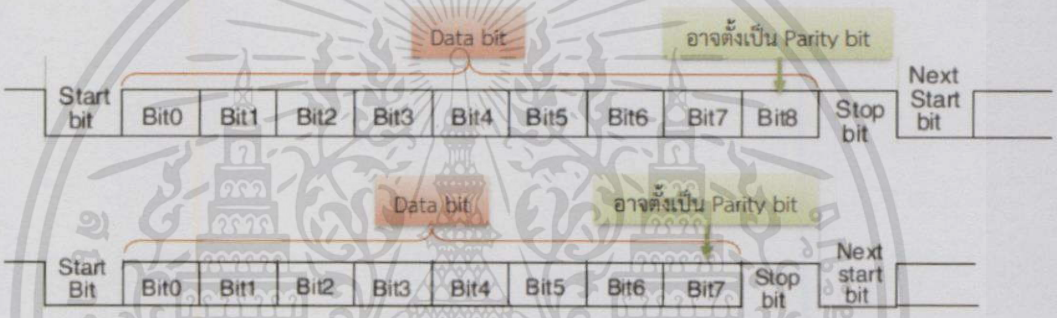
2.2.4 การสื่อสาร

บอร์ดอาร์ดูโน่เมก้านั้นสามารถที่จะสื่อสารกับคอมพิวเตอร์ได้ ซึ่งการสื่อสารกับคอมพิวเตอร์นั้น ATmega2560 จะมี UART สำหรับสื่อสารแบบอนุกรมที่ใช้สัญญาณ TTL (5V) โดยชิพ ATmega16U2 บนบอร์ดจะทำหน้าที่แปลง USB ไปเป็นพอร์ตอนุกรม โดย USB จะไปเชื่อมต่อยังพอร์ตการสื่อสารบนคอมพิวเตอร์ ซึ่งซอฟต์แวร์ที่บรรจุใน Atmega16U2 นั้นใช้ไดรเวอร์การสื่อสาร USB มาตรฐาน โดยซอฟต์แวร์บนคอมพิวเตอร์นั้นจะประกอบไปด้วยจอแสดงผล (serial monitor) ซึ่งทำหน้าที่รับข้อมูลจากบอร์ดอาร์ดูโน่ไปแสดงผล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 การสื่อสารด้วยพอร์ต UART

UART ย่อมาจากคำว่า Universal Asynchronous Receiver Transmitter เป็นการเชื่อมต่อและสื่อสารข้อมูลอนุกรมกับอุปกรณ์ต่าง ๆ เช่น คอมพิวเตอร์, RFID, GPS, GSM Module, Wifi Module เป็นต้น โดยข้อดีของการใช้อะซิงโครนัส (Asynchronous) คือสามารถสื่อสารแบบฟลูดูเพล็กซ์ (Full duplex) ที่สามารถรับและส่งข้อมูลระหว่างตัวรับ (Receiver) และตัวส่ง (Transmitter) ได้ในเวลาเดียวกัน นอกจากนี้ยังไม่จำเป็นต้องใช้สายสัญญาณนาฬิกา เพื่อกำหนดจังหวะการรับส่งข้อมูล แต่มีการกำหนดรูปแบบ หรือ โปรโตคอลการรับส่งข้อมูลขึ้นมาแทน และอาศัยการกำหนดความเร็วของการรับส่งข้อมูลให้เท่ากัน โดยการสื่อสารของ UART ดังแสดงตามรูปที่ 2.2



รูปที่ 2.2 รูปแบบการสื่อสารของ UART

โดยจำเป็นต้องมีการกำหนดคุณสมบัติเหล่านี้ให้เหมือนกันทั้งฝั่งตัวรับและตัวส่ง ซึ่งประกอบด้วย

- บิตเริ่มต้นอยู่ในสถานะต่ำ (Low)
- สามารถกำหนดจำนวนของบิตข้อมูล ให้เป็น 8 หรือ 9 บิต
- สามารถกำหนดชนิดของบิตภาวะ (Parity Bit) ให้เป็น แบบคู่ แบบคี่ หรือ ว่าง
- บิตสิ้นสุดมีจำนวน 0.5, 1, 1.5 หรือ 2 บิต

ระดับแรงดันไฟฟ้าของสัญญาณการสื่อสารแบบ UART สามารถแบ่งได้ดังนี้

- TTL เป็นระดับสัญญาณแบบดิจิตอลทั่วไปที่ใช้กับไมโครคอรโทรลเลอร์ โดยสถานะลอจิก “0” จะมีแรงดันเท่ากับ 0 โวลต์ และสถานะลอจิก “1” จะมีแรงดันเท่ากับ 3.3 หรือ 5 โวลต์
- RS232 เป็นระดับสัญญาณที่ใช้กับคอมพิวเตอร์ โดยสถานะลอจิก “1” จะมีแรงดันเท่ากับ -5 หรือ -13 โวลต์ และสถานะลอจิก “0” จะมีแรงดันเท่ากับ +5 หรือ +13 โวลต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 การสื่อสารข้อมูลด้วย I^2C

I^2C ย่อมาจาก Inter-IC Communication หมายถึง การติดต่อสื่อสารระหว่างไอซี โดยบัส I^2C ได้รับการพัฒนาโดยบริษัท ฟิลิปส์ (Philips) โดยมีจุดมุ่งหมายหลักคือ ต้องการให้ไอซีหรือโมดูลสามารถติดต่อ สั่งงาน และ ควบคุมภายใต้สายสัญญาณเพียง 2 เส้น คือ สายข้อมูลและสายสัญญาณนาฬิกาที่ใช้กำหนดจังหวะการทำงาน โดยการต่อร่วมกันของอุปกรณ์บนบัส I^2C จะต่อสายข้อมูลและสายสัญญาณนาฬิกาของอุปกรณ์แต่ละตัวขนานหรือพ่วงกันไป ส่วนการกำหนดแอดเดรส หรือ กำหนดตำแหน่งสำหรับติดต่ออุปกรณ์แต่ละตัว จะใช้รหัสข้อมูลและการกำหนดสภาวะลอจิกที่ขาแอดเดรสของอุปกรณ์แต่ละตัว สายข้อมูลบนบัส I^2C มีชื่อเรียกอย่างเป็นทางการว่า สายข้อมูลอนุกรม หรือ SDA (Serial Data line) ส่วนสายสัญญาณนาฬิกามีชื่อเรียกว่า สายสัญญาณนาฬิกาอนุกรม หรือ SCL (Serial Clock line) ในการอธิบายต่อไปนี้จะเรียกสายสัญญาณทั้งสองว่า SDA และ SCL

2.4.1 คุณสมบัติโดยทั่วไปของบัส I^2C

ใช้สาย SDA และ SCL เป็นสายสัญญาณ 2 ทิศทาง (bi-directional line) ต้องมีการต่อตัวต้านทานพูลอัพกับแรงดัน +5 โวลต์ ไว้ตลอดเวลา เพื่อให้สายมีสถานะลอจิกสูงในขณะที่ไม่มีการติดต่อใช้งาน ทั้งยังช่วยป้องกันสัญญาณรบกวนที่อาจมีเข้ามาในสายสัญญาณทั้งสอง วงจรเอาท์พุตของอุปกรณ์ที่ต่ออยู่บนบัส I^2C ต้องมีลักษณะเป็นวงจรเดรนเปิด (Open-drain) หรือ คอลเล็กเตอร์เปิด (Open-collector) อัตราการถ่ายทอดข้อมูลบนบัส I^2C สูงถึง 100 กิโลบิตต่อวินาทีในโหมดปกติ และ สูงถึง 400 กิโลบิตต่อวินาทีในโหมดความเร็วสูง อุปกรณ์ที่ต่อร่วมบนบัส I^2C จะต้องมีความจุไฟฟ้ารวมที่เกิดขึ้นระหว่างสาย SDA และ SCL ไม่เกิน 400 พิโกฟารัด (pF) การเข้าถึงอุปกรณ์บนบัส I^2C ใช้ข้อมูลสำหรับการเข้าถึงสองค่าคือ 7 บิต (7-bit addressing) หรือ 10 บิต (10-bit addressing) ข้อดีอีกประการของบัส I^2C คือ สามารถเชื่อมต่ออุปกรณ์ที่ใช้ไฟเลี้ยง +5 โวลต์ ในขณะที่อีกตัวหนึ่งใช้ไฟเลี้ยง +12 โวลต์ การต่อร่วมกันบนบัส I^2C สามารถทำได้ในลักษณะเดียวกันกับกรณีที่อุปกรณ์ทั้งสองใช้ไฟเลี้ยงเท่ากันกล่าวคือให้ต่อสาย SDA และ SCL ของอุปกรณ์แต่ละตัวเข้าด้วยกัน และต้องต่อตัวต้านทานพูลอัพ (Rp) เข้ากับแรงดัน +5 โวลต์ ไว้ด้วยเสมอ

2.4.2 หลักการของบัส I^2C

บัส I^2C ประกอบด้วยสายสัญญาณ 2 เส้นคือ SDA และ SCL อุปกรณ์ที่ต่อพ่วงบนบัสสามารถมีได้มากมาย ดังนั้นจึงต้องมีการกำหนดรูปแบบของการติดต่อบนบัส เพื่อให้ผู้ใช้งานทราบว่าขณะนี้อุปกรณ์ใดติดต่อกันอยู่ และอุปกรณ์ใดเป็นตัวรับหรือส่ง โดยสามารถอธิบายลักษณะ หน้าที่ และนิยามของอุปกรณ์บนบัส I^2C ได้ดังนี้ เพื่อเป็นข้อตกลงก่อนอธิบายการทำงานของบัส I^2C ต่อไป อุปกรณ์ที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เป็นผู้สร้างข้อมูลหรือส่งข้อมูลเรียกว่าตัวส่ง (transmitter) อุปกรณ์ที่เป็นผู้รับข้อมูล เรียกว่า ตัวรับ (receiver) อุปกรณ์บนบัส I^2C สามารถเป็นได้ทั้งตัวรับและส่ง บางอุปกรณ์ทำหน้าที่เป็นตัวรับอย่างเดียว จะไม่มีอุปกรณ์ใดบนบัส I^2C ที่ทำหน้าที่เป็นตัวส่งอย่างเดียว อุปกรณ์ที่ทำหน้าที่ควบคุมจังหวะการติดต่อบนบัส I^2C เรียกว่า มาสเตอร์ (master) อุปกรณ์ที่ถูกควบคุมหรืออุปกรณ์ที่ต่อพ่วงเข้าไปบนบัส I^2C เรียกว่า สเลฟ (slave) ข้อกำหนด 2 ประการสำคัญของการติดต่อบนบัส I^2C คือ

1. การถ่ายทอดข้อมูลจะเกิดขึ้นได้เมื่อบัสว่างเท่านั้น
2. ในระหว่างการถ่ายทอดข้อมูล เมื่อใดก็ตามที่สาย SCL มีสถานะลอจิกสูง สายข้อมูลต้องรักษาข้อมูลไว้

2.4.3 สภาวะที่เกิดขึ้นบนบัส I^2C

- บัสว่าง สภาวะนี้เกิดขึ้นเมื่อ สถานะลอจิกบนสาย SDA และ SCL มีลอจิกสูงทั้งคู่นั้น หมายความว่า การถ่ายทอดข้อมูลสามารถเริ่มต้นขึ้นได้
- การเริ่มต้นการถ่ายทอดข้อมูล เกิดขึ้นเมื่อสาย SDA มีการเปลี่ยนแปลงลอจิกจากสูงไปต่ำ ในขณะที่สาย SCL มีสถานะลอจิกสูง เรียกสภาวะนี้ว่า สภาวะเริ่มต้น
- การหยุดการถ่ายทอดข้อมูลเกิดขึ้นเมื่อสาย SDA มีการเปลี่ยนแปลงระดับลอจิกจากต่ำไปสูง ในขณะที่สาย SCL มีสถานะลอจิกสูงเรียกสภาวะที่เกิดขึ้นนี้ว่าสภาวะหยุด
- ข้อมูลดำรงอยู่บนบัส สภาวะนี้เกิดขึ้นถัดจากสภาวะเริ่มต้น โดยสถานะลอจิกที่เกิดขึ้นบนสาย SDA ก็คือข้อมูลที่ทำการถ่ายทอด เมื่อสาย SCL มีลอจิกสูง สถานะที่สาย SDA ต้องคงที่ เพื่อให้อุปกรณ์รับข้อมูลในจังหวะนั้นเป็น "0" หรือ "1" ข้อมูลอาจเกิดการเปลี่ยนแปลงได้ในขณะที่สาย SCL เป็นลอจิกต่ำ แต่เมื่อใดก็ตามที่ต้องการให้เกิดการถ่ายทอดข้อมูลอย่างสมบูรณ์ สถานะลอจิกที่ขา SDA ต้องคงที่ตลอดช่วงเวลาที่สาย SCL มีสถานะลอจิกสูง หากเกิดการเปลี่ยนแปลงสถานะลอจิกในขณะที่สาย SCL มีลอจิกสูงอยู่นั้น อุปกรณ์มาสเตอร์ที่ควบคุมการถ่ายทอดข้อมูลจะแปลความหมายเป็นสภาวะหยุด หรือ สภาวะเริ่มต้นก็ได้ ทำให้ข้อมูลที่ทำการถ่ายทอดเกิดความผิดพลาดเกิดขึ้น
- การรับรู้ข้อมูล เกิดขึ้นหลังจากการถ่ายทอดข้อมูลจากตัวส่งมายังตัวรับเกิดขึ้นอย่างสมบูรณ์ โดยตัวส่งจะทำการส่งข้อมูลมา 1 บิตเรียกว่า บิตรับรู้ (acknowledge bit) มีสถานะเป็นลอจิกสูง หลังการส่งข้อมูลมาครบถ้วน ส่วนอุปกรณ์มาสเตอร์จะทำการส่งสัญญาณรับรู้พิเศษซึ่งสัมพันธ์กับสัญญาณนาฬิกา อุปกรณ์สเลฟที่ถูกอ้างอิงในการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ติดต่อ หรือ กำลังติดต่ออยู่ในขณะนั้นก็จะกำเนิดบิตรับรู้ที่มีสถานะลอจิกต่ำ เพื่อตอบสนองให้ทราบว่า ได้รับข้อมูลเรียบร้อยแล้ว

2.4.4 การทำงานบนบัส I^2C

ก่อนเริ่มต้นการถ่ายทอดข้อมูลระหว่างอุปกรณ์ต่าง ๆ ที่ต่ออยู่บนบัส ต้องเข้าถึงอุปกรณ์เสียก่อน โดยการเข้าถึงอุปกรณ์บนบัส I^2C นั้นจะใช้การเข้าถึงแบบ 7 หรือ 10 บิต ในกรณีที่มีอุปกรณ์ต่ออยู่บนบัสไม่มาก ใช้การเข้าถึงแบบ 7 บิตก็เพียงพอ แต่ในบางอุปกรณ์ต้องใช้การเข้าถึงแบบ 10 บิต หลังจากที่ติดต่ออุปกรณ์แต่ละตัวเรียบร้อยแล้ว ก็จะเริ่มต้นการถ่ายทอดข้อมูลกันต่อไป ดังนั้นหัวใจสำคัญในอันดับแรกของการทำงานบนบัส I^2C คือการอ้างถึงอุปกรณ์แต่ละตัวซึ่งในที่นี้จะอธิบายรายละเอียดของการอ้างถึงทั้ง 2 รูปแบบ

2.4.4.1 การอ้างอิงถึงแบบ 7 บิต (7-bit addressing)

ข้อมูลไบต์แรกที่เกิดขึ้นหลังจากสภาวะเริ่มต้นคือ ข้อมูลที่ใช้อ้างถึงอุปกรณ์ที่ต้องการติดต่อ ใน 7 บิตบน รวมทั้งบิตที่มีนัยสำคัญต่ำสุด (LSB) ด้วยจะเป็นข้อมูลแอดเดรสของอุปกรณ์สเลฟที่ต้องการติดต่อ โดยแบ่งเป็น บิตกำหนดแอดเดรสคงที่ (fix address bit) จำนวน 4 บิต ซึ่งข้อมูลนี้อุปกรณ์แต่ละตัวจะถูกกำหนดมาจากผู้ผลิต ไม่สามารถเปลี่ยนแปลงแก้ไขได้ ถัดมาอีก 3 บิตเป็นบิตกำหนดแอดเดรสที่สามารถโปรแกรมได้ (programmable address bit) โดยผู้ใช้งานต้องกำหนดสถานะลอจิกให้แก่ขา A0-A2 ของอุปกรณ์ที่มีการเชื่อมต่อแบบบัส I^2C ส่วนในบิตที่มีนัยสำคัญต่ำสุดที่ใช้กำหนดการอ่านหรือเขียนข้อมูลกับอุปกรณ์สเลฟตัวนั้น ๆ หากบิตที่มีนัยสำคัญต่ำสุด (LSB) เป็น "0" หมายถึงต้องการเขียนข้อมูลไปยังอุปกรณ์นั้น ถ้าเป็น "1" จะเป็นการอ่านข้อมูลจากอุปกรณ์สเลฟ ข้อมูลในไบต์ต่อมาคือ ข้อมูลควบคุม (control byte) ในอุปกรณ์แต่ละตัวจะมีการกำหนดข้อมูลควบคุมที่แตกต่างกันไป ยกตัวอย่าง ไอซีขยายพอร์ตมีข้อมูลควบคุมที่ใช้กำหนดว่าบิตใดเป็นอินพุต บิตใดเป็นเอาท์พุต ในขณะไอซี ADC/DAC ต้องการข้อมูลควบคุมเพื่อกำหนดให้ทำงานเป็นวงจร ADC หรือ DAC เป็นต้น ข้อมูลในไบต์ต่อมาคือ ข้อมูลที่ทำการถ่ายทอดจริง (data) หลังจากการถ่ายทอดข้อมูลในแต่ละไบต์ อุปกรณ์สเลฟที่ได้รับการติดต่อต้องส่งสัญญาณรับรู้ตอบกลับมาด้วยทุกครั้ง เพื่อให้กระบวนการถ่ายทอดข้อมูลสามารถดำเนินต่อไปได้

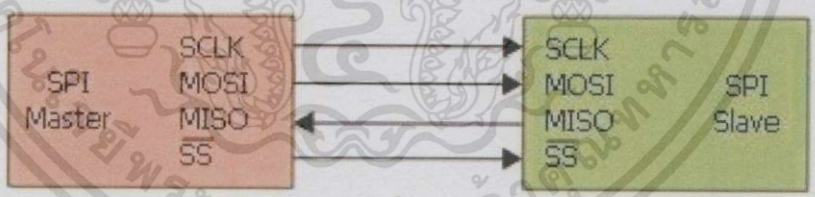
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.4.2 การเข้าถึงแบบ 10 บิต (10-bit addressing)

ในการอ้างถึงแบบนี้ยังคงใช้รูปแบบข้อมูลเหมือนกับแบบ 7 บิต โดยจะมีข้อมูลเพิ่มเติมขึ้นมาเล็กน้อย โดยในไบต์แรกหลังจากสภาวะเริ่มต้น ต้องกำหนดให้ 5 บิตบนมีข้อมูลเป็น 11110 ส่วนอีก 2 บิตถัดมาเป็นบิตแอดเดรสของอุปกรณ์ที่ต้องการติดต่อ ในบิตที่มีนัยสำคัญต่ำสุด (LSB) ของข้อมูลไบต์แรก ยังคงเป็นการกำหนดว่าต้องการอ่านหรือเขียนข้อมูลกับอุปกรณ์สเลฟตัวที่ต้องการติดต่อด้วย ข้อมูลไบต์ที่ 2 เป็นข้อมูลแอดเดรสของอุปกรณ์ที่ต้องการติดต่อด้วย ข้อมูลไบต์ถัดไปจึงเป็นข้อมูลควบคุม ข้อมูลหลังจากนี้ก็จะเป็นข้อมูลจริงที่ใช้ในการติดต่อ เช่นเดียวกันกับการเข้าถึงแบบ 7 บิต หลังจากถ่ายทอดข้อมูลครบทุกไบต์ ต้องมีสภาวะรับรู้เกิดขึ้น เพื่อให้ขบวนการถ่ายทอดข้อมูลสามารถดำเนินต่อไปได้

2.5 การสื่อสารข้อมูลด้วย SPI

SPI หรือ Serial Peripheral Interface เป็นการสื่อสารอนุกรมแบบซิงโครนัสอีกรูปแบบหนึ่ง ซึ่งทำงานในรูปแบบที่ให้อุปกรณ์ตัวหนึ่งทำหน้าที่เป็นมาสเตอร์ ในขณะที่อีกตัวหนึ่งทำหน้าที่เป็นสเลฟ ดังแสดงในรูปที่ 2.3 โดยสามารถส่งข้อมูลในโหมดฟลูดูเพล็กซ์ คือสัญญาณสามารถส่งหากันระหว่างมาสเตอร์ และสเลฟได้อย่างต่อเนื่อง รูปแบบข้อมูลการสื่อสารหรือโปรโตคอลของการสื่อสารแบบ SPI นี้ไม่ได้มีมาตรฐานกำหนดตายตัว ว่าข้อมูลที่ส่งหากันต้องอยู่ในรูปแบบใด เป็นการคิดโปรโตคอลการสื่อสารกันเอาเอง หรือดูจากหนังสือข้อมูลของอุปกรณ์ประกอบ



รูปที่ 2.3 การเชื่อมต่อการสื่อสารแบบ SPI ระหว่างอุปกรณ์ มาสเตอร์-สเลฟ

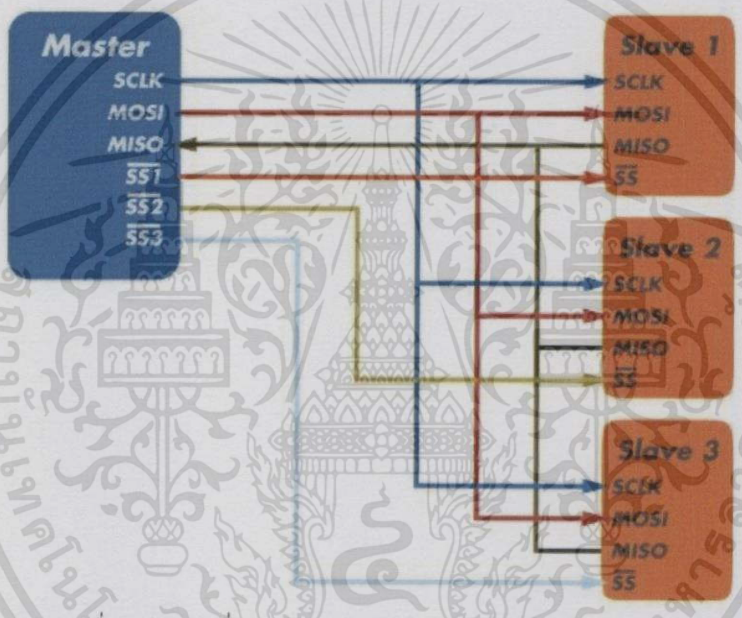
การเชื่อมต่อการสื่อสารแบบ SPI ระหว่างอุปกรณ์มาสเตอร์และสเลฟ มีสายสัญญาณสี่เส้น (Four Wire) ประกอบด้วย

1. SCLK (Serial Clock) ใช้ส่งสัญญาณนาฬิกาจากอุปกรณ์มาสเตอร์ไปยังอุปกรณ์สเลฟเพื่อกำหนดจังหวะการรับส่งข้อมูล
2. MOSI (Master Out Slave In) ใช้ส่งข้อมูลจากอุปกรณ์มาสเตอร์ไปยังอุปกรณ์สเลฟ
3. MISO (Master In Slave Out) ใช้รับข้อมูลจากอุปกรณ์สเลฟ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. SS (Slave Select) หรือ ขา CS (Chip Select) ใช้ส่งสัญญาณสถานะต่ำ (Low) ไปยังอุปกรณ์สเลฟที่ต้องการรับส่งข้อมูล

โดยอุปกรณ์มาสเตอร์ทำหน้าที่เป็นตัวควบคุมการสื่อสารทั้งหมด โดยควบคุมการสื่อสารตามสัญญาณนาฬิกา และสายสัญญาณ SS โดยตัวมาสเตอร์จะเป็นตัวตัดสินใจเลือก รับ หรือ ส่ง ข้อมูลไปยังอุปกรณ์สเลฟ ในกรณีที่มีตัวสเลฟมากกว่า 1 ตัว หากต้องการสื่อสารกับสเลฟตัวใด ต้องทำให้เส้น SS ของสเลฟตัวนั้นมีระดับสัญญาณอยู่ในสถานะต่ำ ดังรูปที่ 2.4 ซึ่งแสดงการเชื่อมต่อมาสเตอร์กับอุปกรณ์สเลฟหลายตัว



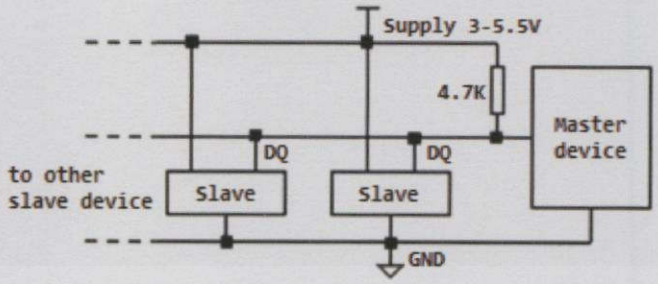
รูปที่ 2.4 การเชื่อมต่อการสื่อสารแบบ SPI ระหว่างอุปกรณ์มาสเตอร์และสเลฟหลายตัว

2.6 การสื่อสารด้วยสายสัญญาณ 1 เส้น (1-wire Communication)

การสื่อสารแบบสายสัญญาณ 1 เส้น เป็นระบบบัสข้อมูลแบบฮาร์ฟดูเพล็กซ์ คือสามารถสื่อสารได้ 2 ทิศทาง แต่ไม่สามารถรับและส่งข้อมูลในช่วงเวลาเดียวกันได้ ระบบบัสมีการทำงานเป็นแบบมาสเตอร์/สเลฟ โดยอุปกรณ์มาสเตอร์จะเป็นตัวควบคุมสถานะและจังหวะการรับส่งของบัสข้อมูล ในขณะที่อุปกรณ์สเลฟ จะทำงานตามการควบคุมของอุปกรณ์มาสเตอร์เท่านั้น

ในการใช้งานบัสแบบสายสัญญาณ 1 เส้นนี้ สายสัญญาณข้อมูล DQ จะต้องมีสภาวะปกติที่ลอจิกสูง สามารถทำได้โดยการต่อตัวต้านทานประมาณ 5 กิโลโอห์มพูลอัพไว้กับไฟเลี้ยง หรือในกรณีที่ใช้บัสสายสัญญาณ 1 เส้นต่ออุปกรณ์มาสเตอร์ร่วมกับสเลฟหลายตัว ก็สามารถทำได้ดังแสดงในรูปที่ 2.5

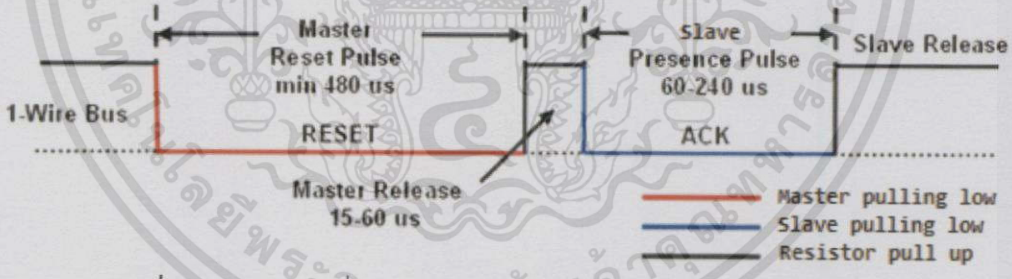
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.5 การใช้บัสสายสัญญาณ 1 เส้นระหว่างอุปกรณ์มาสเตอร์และสเลฟหลายตัว

2.6.1 การต่อใช้งานระหว่างอุปกรณ์มาสเตอร์และสเลฟ

รูปแบบของสัญญาณบนบัสสายสัญญาณ 1 เส้น สามารถแบ่งออกได้เป็น 6 รูปแบบ คือ สัญญาณเริ่มต้น (Reset pulse) สัญญาณตอบสนอง (Presence pulse) เขียนข้อมูล 0 (write 0) เขียนข้อมูล 1 (write 1) อ่านข้อมูล 0 (read 0) และ อ่านข้อมูล 1 (read 1) ซึ่งในกระบวนการเริ่มต้นการสื่อสารสายสัญญาณ 1 เส้นทั้งหมดนั้น อุปกรณ์มาสเตอร์ต้องส่งสัญญาณร้องขอเริ่มการสื่อสารด้วยการสร้างสัญญาณเริ่มต้นก่อน เมื่ออุปกรณ์สเลฟได้รับสัญญาณเริ่มต้นก็จะสร้างสัญญาณตอบสนอง เพื่อตอบรับการขอเริ่มการสื่อสารนั้น ซึ่งมีรายละเอียดของช่วงเวลาต่าง ๆ ดังแสดงในรูปที่ 2.6

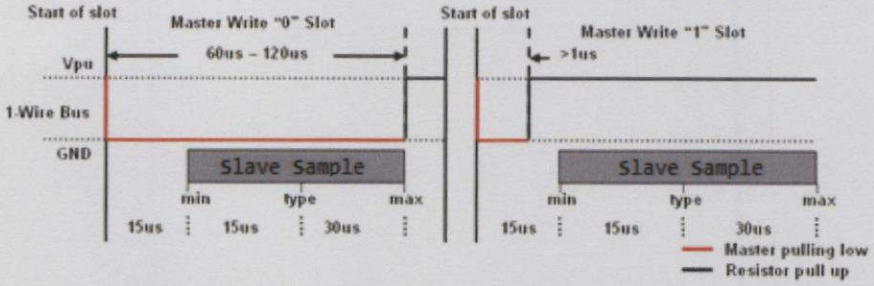


รูปที่ 2.6 การติดต่อสื่อสารแบบสายสัญญาณ 1 เส้นกับอุปกรณ์สเลฟ

2.6.1.1 การเขียนข้อมูลไปยังอุปกรณ์สเลฟ

ในการเขียนข้อมูลแบ่งออกเป็น 2 ชนิดคือการเขียนข้อมูล “1” และการเขียนข้อมูล “0” ดังแสดงในรูปที่ 2.7 การเขียนข้อมูลไปยังอุปกรณ์สเลฟต้องใช้ช่วงเวลาอย่างต่ำ 60 ไมโครวินาที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

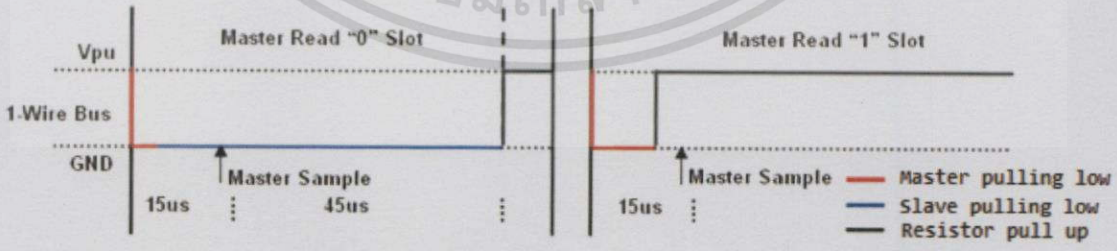


รูปที่ 2.7 การเขียนข้อมูลไปยังอุปกรณ์สเลฟ

การเขียนข้อมูลทั้ง 2 ชนิดนั้น เริ่มแรกอุปกรณ์มาสเตอร์ต้องดึงสัญญาณบนบัสสายสัญญาณ 1 เส้นลงมาให้อยู่ในสถานะลอจิกต่ำก่อน ในกรณีที่ต้องการเขียนข้อมูล “0” ลงไปยังอุปกรณ์สเลฟ อุปกรณ์มาสเตอร์ต้องดึงสัญญาณบนบัสให้เป็นลอจิกต่ำต่อจนกว่าจะครบคาบเวลา (Time slot) อย่างต่ำ 60 ไมโครวินาที ส่วนในกรณีที่ต้องการเขียนข้อมูล “1” ไปยังอุปกรณ์สเลฟ อุปกรณ์มาสเตอร์ต้องปล่อยบัส เพื่อให้บัสกลับไปอยู่ในสถานะลอจิกสูงก่อนการสุ่มของอุปกรณ์สเลฟ ซึ่งจะอยู่ในช่วง 15-60 ไมโครวินาที หลังจากที่อุปกรณ์มาสเตอร์ดึงสัญญาณบัสสายสัญญาณ 1 เส้น ลงมา

2.6.1.2 การอ่านข้อมูลจากอุปกรณ์สเลฟ

ในการอ่านค่าภายในหน่วยความจำชั่วคราว (SRAM) ของอุปกรณ์สเลฟนั้นสามารถทำได้ ก็ต่อเมื่ออุปกรณ์มาสเตอร์ได้เขียนข้อมูลเพื่อขอทำการอ่านค่าในหน่วยความจำชั่วคราว (Read Scratchpad) ซึ่งมีค่าเป็น 0xBE ลงไปที่อุปกรณ์สเลฟเสียก่อน จากนั้นจึงเริ่มอ่านข้อมูลจากบัสสัญญาณ 1 เส้น โดยคาบเวลา (Time slot) ของการอ่านต้องมีช่วงเวลาอย่างต่ำ 60 ไมโครวินาที และต้องมีช่วงเวลาระหว่างคาบเวลาอย่างต่ำ 1 ไมโครวินาที ดังแสดงในรูปที่ 2.8



รูปที่ 2.8 การอ่านข้อมูลจากบัสสัญญาณ 1 เส้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การอ่านข้อมูลจากบัสสัญญาณ 1 เส้น เริ่มแรกอุปกรณ์มาสเตอร์จะต้องดึงบัสสัญญาณ 1 เส้นลงให้อยู่ในสถานะลอจิกต่ำเป็นช่วงเวลาอย่างน้อย 1 ไมโครวินาที จากนั้นจึงค่อยปล่อยบัส ในกรณีอุปกรณ์สเลฟส่งข้อมูล “0” อุปกรณ์สเลฟจะดึงบัสให้เป็นลอจิกต่ำจนกว่าจะสิ้นสุดไทม์สล็อตจึงจะปล่อยให้บัสกลับไปอยู่ในสถานะลอจิกสูง ส่วนในกรณีที่อุปกรณ์สเลฟส่งข้อมูล “1” อุปกรณ์สเลฟจะปล่อยบัสให้อยู่ในสภาวะลอจิกสูงตลอด โดยในการสุ่มเพื่อรับข้อมูลจากอุปกรณ์สเลฟควรทำใน 15 ไมโครวินาที หลังจากจุดเริ่มของคาบเวลา (Time slot)

2.7 หน้าจอระบบสัมผัส

หน้าจอระบบสัมผัสถูกออกแบบและผลิตโดยบริษัท 4Dsystem ซึ่งเป็นผู้นำในการวิจัยและพัฒนาเกี่ยวกับการใช้เทคโนโลยีกราฟิก โดยมีเอกลักษณ์และมีประสิทธิภาพที่ดี บริษัทก่อตั้งขึ้นในปี 1990 โดยประสบการณ์ต่าง ๆ นั้น ช่วยผลักดันให้ก้าวข้ามข้อจำกัดด้านต่าง ๆ ทั้งอุปกรณ์รวมถึงซอฟท์แวร์ สามารถทำงานสอดคล้องกันได้ตามความต้องการของผู้ใช้ เหมาะสมกับการนำมาใช้งานในหลาย ๆ ประเภท

2.7.1 โปรแกรมสำหรับออกแบบหน้าจอระบบสัมผัส

โปรแกรม Workshop4 นั้นคือซอฟต์แวร์ที่เป็นเครื่องมือในการพัฒนาและออกแบบใช้งานหน้าจอระบบสัมผัส ที่ครอบคลุมการใช้งานบนระบบปฏิบัติการวินโดว์ (Windows) ซึ่งสามารถที่จะเลือกรูปแบบของการออกแบบได้หลากหลาย ประกอบไปด้วยรูปแบบ Designer Visi ViSi-Genie และ Serial โดยตัวที่ใช้ในโครงงานนี้คือ ViSi-Genie ซึ่งเป็นโปรแกรมในรูปแบบของการออกแบบหน้าจอระบบสัมผัส โดยมีกระบวนการที่ง่าย เนื่องจากไม่ต้องมีการเขียนโปรแกรมการทำงาน แต่โปรแกรมการทำงานจะถูกเขียนขึ้นเองอัตโนมัติโดยซอฟต์แวร์ จึงทำให้ผู้ใช้สามารถสร้างจอภาพ (User interface) จากแบบร่างได้ในเวลาอันรวดเร็ว โดยแต่ละวัตถุนั้นสามารถที่จะกำหนดเหตุการณ์เพื่อให้ทำงาน รวมถึงกำหนดตำแหน่ง ขนาด และตัวแปรอื่นๆ เช่น สี ช่วงของข้อมูล ตัวอักษร เป็นต้น รวมถึงการกำหนดวัตถุให้ทำงานหรือสื่อสารกับอุปกรณ์ภายนอก เช่น อาร์ดูโน่ ดังแสดงในรูปที่ 2.9 โดยวัตถุนั้นสามารถแบ่งออกเป็น 3 กลุ่ม ดังนี้

- วัตถุรับค่า (Input object)

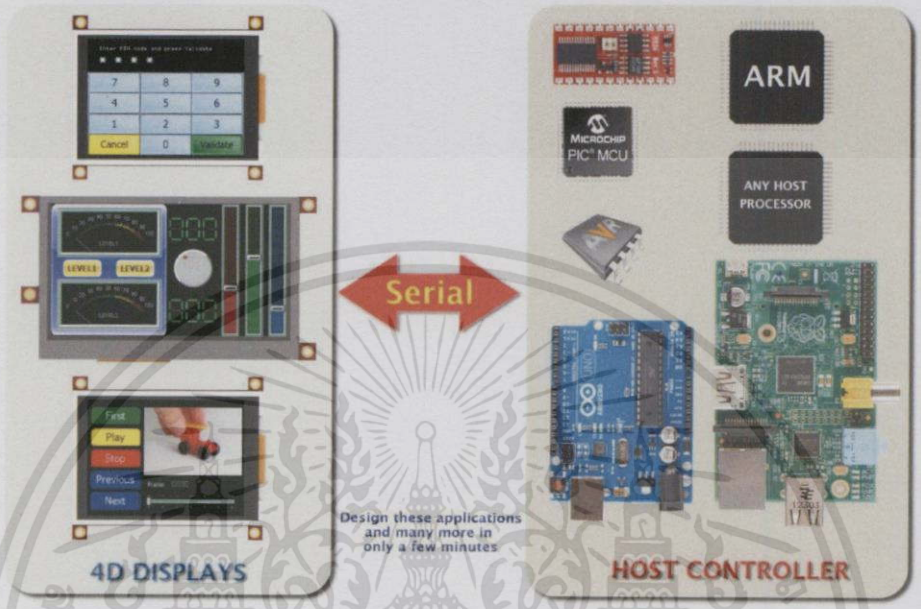
  วัตถุรับค่า นั้นคือ ปุ่มหรือคีย์บอร์ด
- วัตถุแสดงค่า (output object)

  วัตถุแสดงค่า เช่น เกจวัด หรือ มิเตอร์ต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- วัตถุทั้งรับและแสดงค่า (combined object, input/output object)

  วัตถุทั้งรับและแสดงค่า นั้นเป็นวัตถุที่จะเป็นทั้งส่วนอินพุทและส่วนเอาท์พุท



รูปที่ 2.9 การเชื่อมต่อระหว่างจอสัมผัสกับโฮสต์

2.8 อุปกรณ์บันทึกข้อมูล (Datalogger shield)

อุปกรณ์บันทึกข้อมูลลงบนการ์ดหน่วยความจำแสดงตามรูปที่ 2.10 สามารถใช้งานได้ทั้งแบบแฟต16 (FAT16) และ แฟต32 (FAT32) โดยในโมดูลนี้จะมีส่วนของ ค่าเวลานาฬิกาจริง (Real Time Clock ; RTC) ที่จะระบุข้อมูลพร้อมวันเวลา ณ ช่วงที่ทำการบันทึกข้อมูลรวมอยู่ด้วย จึงสามารถที่จะรู้ได้ว่าข้อมูลที่ได้รับนั้นเกิดขึ้น ณ วันเวลาใด

ในความจริงแล้วภายในอาร์ดูโน่นั้น สามารถที่จะเขียนโปรแกรมเรียกใช้งานฟังก์ชันจับเวลา (millis()) เพื่อใช้บอกเวลาได้เช่นกัน แต่เนื่องจากฟังก์ชันจับเวลาที่เรียกใช้งานนั้น จะเริ่มจับเวลาตั้งแต่อาร์ดูโน่ได้รับพลังงาน และเวลาจะกลับไปเป็นศูนย์เมื่อไม่มีพลังงาน ทำให้ต้องเริ่มต้นจับเวลาใหม่อยู่เสมอ นอกจากนี้อาร์ดูโน่ยังไม่สามารถบอกวันหรือเดือนได้ เช่น วันอังคาร วันจันทร์ หรือ เดือนมีนาคม เดือนเมษายน ได้ แต่จะบอกได้เพียงจำนวนระยะเวลาที่อาร์ดูโน่ทำงานไปแล้วเท่านั้น เช่น 14,000 มิลลิวินาที เป็นต้น

ดังนั้นจึงต้องมีชิพค่าเวลานาฬิกาจริง (RTC) แยกออกมาและได้รับพลังงานจากอีกแหล่งจ่ายนึงตลอดเวลา เพื่อที่จะไม่ต้องตั้งเวลาใหม่ทุกครั้งที่มีการใช้งาน โดยอุปกรณ์บันทึกข้อมูลนั้นถูกออกแบบมาให้มีความสามารถที่หลากหลาย และสามารถที่จะนำไปดัดแปลงได้ง่าย

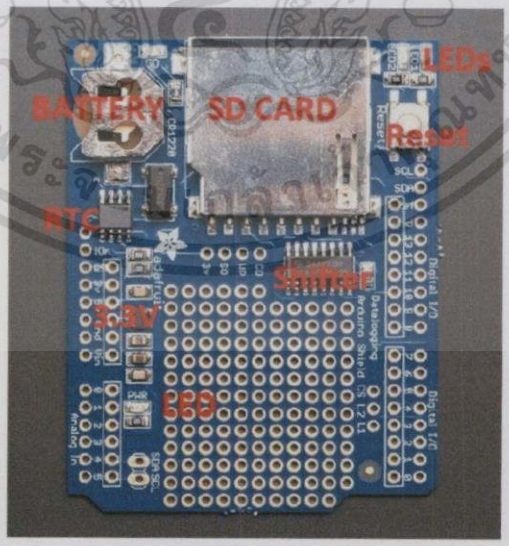
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.8.1 คุณสมบัติของอุปกรณ์บันทึกข้อมูล

ระบบจัดการไฟล์ของการ์ดหน่วยความจำนั้นเป็นได้ทั้งแบบ แฟต16 และ แฟต32 โดยมีวงจรปรับระดับแรงดันเพื่อป้องกันอันตรายที่จะเกิดขึ้นกับการ์ดหน่วยความจำ และมีชิพค่าเวลานาฬิกาจริง (RTC) โดยแบตเตอรี่สามารถใช้ในการสำรองพลังงานได้หลายปี อีกทั้งอุปกรณ์ยังมีพื้นที่สำหรับการบัดกรีวงจรหรืออุปกรณ์ตรวจรู้ต่าง ๆ

2.8.2 ส่วนประกอบหลักภายในอุปกรณ์บันทึกข้อมูล

1. RTC (real time clock) ซึ่งประกอบไปด้วยชิพ DS1307 คริสตัล และ แบตเตอรี่สำรองซึ่งทำงานได้ยาวนานถึง 7 ปี
2. ตัวควบคุมระดับแรงดัน (Regulator) ทำหน้าที่ช่วยให้อุปกรณ์บันทึกข้อมูลสามารถทำงานได้อย่างมีเสถียรภาพที่ระดับความดัน 3 โวลต์
3. การ์ดหน่วยความจำสามารถที่จะรองรับได้ทั้งแบบ SD และ MMC ที่ความจุสูงถึง 32 กิ๊กกะไบต์ แต่ในกรณีของการ์ดหน่วยความจำขนาดเล็ก (MicroSD card) นั้น จำเป็นจะต้องมีตัวแปลงสำหรับแปลงขนาด เพื่อให้ใส่เข้าไปในช่องการ์ดหน่วยความจำได้
4. วงจรยกระดับแรงดัน (Shifter) ทำหน้าที่ป้องกันความเสียหายของการ์ดหน่วยความจำของสัญญาณแรงดัน 5 โวลต์ จากอาร์ดูโน่ เนื่องจากชิลล์ทำงานที่ระดับแรงดัน 3 โวลต์ ดังรูปที่ 2.10



รูปที่ 2.10 ส่วนประกอบภายในอุปกรณ์บันทึกข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.9 เซ็นเซอร์วัดอุณหภูมิ

เซ็นเซอร์วัดอุณหภูมิ หรือ DS18B20 ดังแสดงในรูปที่ 2.11 เป็นไอซีวัดอุณหภูมิแบบดิจิตอลของบริษัท Dallas Semiconductor สามารถวัดอุณหภูมิเป็นหน่วยองศาเซลเซียส ในช่วง -55 ถึง 125 อาศาเซลเซียส ที่ความละเอียด 9-10 บิต และมีความแม่นยำอยู่ที่ 0.5 องศาเซลเซียส ในช่วง -10 ถึง 85 องศาเซลเซียส

การสื่อสารและควบคุม DS18B20 นั้นสามารถทำได้โดยใช้บัสข้อมูลแบบสัญญาณ 1 เส้นของ Dallas Semiconductor ซึ่งใช้สายสัญญาณเพียงแค่เส้นเดียวเท่านั้น ภายใน DS18B20 แต่ละตัวจะมีรหัสประจำตัวขนาด 64 บิต ทำให้สามารถใช้งานพร้อมกันได้หลายตัวบนบัสแบบสัญญาณ 1 เส้น นอกจากนี้ยังสามารถทำงานในโหมดใช้ไฟเลี้ยงจากตัวต้านทานพลูอัพ (Parasite Power Mode) ซึ่งเป็นการทำงานแบบไม่ต้องใช้ไฟเลี้ยงแต่ใช้พลังงานจากสายสัญญาณ 1 เส้น (1 wire communication) ซึ่งมีประโยชน์มากสำหรับการวัดอุณหภูมิระยะไกล หรือในการใช้ในที่ที่มีเนื้อที่จำกัด



ขา	สัญลักษณ์	อธิบาย
1	GND	กราวนด์
2	DQ	สายข้อมูล
3	V_{dd}	ไฟเลี้ยง

รูปที่ 2.11 โครงสร้าง และขาของ DS18B20

ข้อมูลอุณหภูมิที่วัดได้จะถูกเก็บอยู่ในส่วนบันทึกข้อมูลอุณหภูมิ (Register) ซึ่งมีขนาด 16 บิต ในกรณีที่ DS18B20 ทำงานในโหมดความละเอียด 12 บิต บิตทุกบิตในส่วนบันทึกข้อมูลอุณหภูมิจะถูกใช้หมด แต่ในกรณี 9–11 บิต บิตล่าง (บิต0–บิต2) จะไม่ถูกใช้งาน สามารถกำหนดความละเอียดของบิตด้วยค่าที่ใช้กำหนดส่วนบันทึกข้อมูล (Register Configuration)

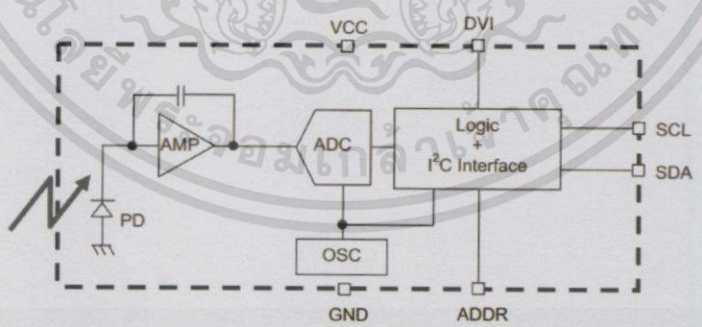
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.10 เซ็นเซอร์วัดความสว่าง

BH 1750 FVI เป็นเซ็นเซอร์วัดความสว่างโดยรอบแบบดิจิตอลซึ่งใช้การเชื่อมต่อแบบ I^2C bus interface ในการเชื่อมต่อกับอุปกรณ์อื่นๆ ไอซีนี้เป็นตัวที่เหมาะสำหรับการวัดความสว่างโดยรอบเพื่อปรับความสว่างของหน้าจอและแสงที่แป้นพิมพ์ของโทรศัพท์มือถือซึ่ง BH 1750 มีรัศมีการวัดกว้างและมีความละเอียดที่สูงถึง 65,532 ลักซ์ แสดงวงจรภายในได้ดังรูปที่ 2.12

2.10.1 คุณสมบัติของ BH 1750 FVI

1. ใช้การเชื่อมต่อแบบ I^2C
2. เสถียรภาพการตอบสนองประมาณสายตามนุษย์
3. แปลงค่าความสว่างเป็นดิจิตอลเอาท์พุต
4. รัศมีการวัดกว้าง และมีความละเอียดสูง (1 - 65535 ลักซ์)
5. ใช้พลังงานน้อย โดยใช้โหมดประหยัดพลังงาน
6. มีฟังก์ชันการกรองแสงรบกวนความถี่ 50/60 Hz
7. ไม่จำเป็นต้องมีอุปกรณ์เพิ่มเติม
8. สามารถปรับค่าผลของการวัดสำหรับนำไปแสดงผล (วัดได้น้อยที่สุด 0.11 ลักซ์ และมากที่สุด 100000 ลักซ์ โดยปรับที่ฟังก์ชันการทำงาน)
9. ค่าการวัดเปลี่ยนแปลงน้อย (+/- 20%)
10. ได้รับอิทธิพลจากอินฟราเรดเพียงเล็กน้อย



รูปที่ 2.12 วงจรภายในของ BH 1750 FVI

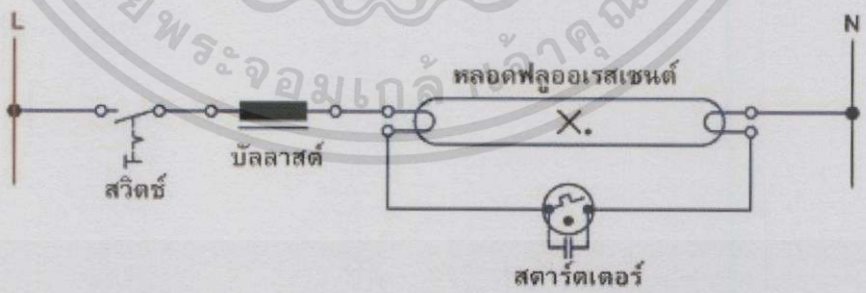
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.11 หลอดฟลูออเรสเซนต์ (Fluorescent Lamp)

ทำด้วยหลอดแก้วที่สูบอากาศออกจนหมด แล้วบรรจุไอปรอทไว้เล็กน้อย มีไส้ที่ปลายหลอดทั้งสองข้าง โดยหลักการทำงานของหลอดฟลูออเรสเซนต์นั้น คือเมื่อกระแสไฟฟ้าผ่านไอปรอทจะคายพลังงานไฟฟ้าให้อะตอมของไอปรอท ทำให้อะตอมของไอปรอทอยู่ในสภาวะถูกกระตุ้น (excited state) และอะตอมของไอปรอทจะคายพลังงานออกมาเพื่อลดระดับพลังงานในรูปของรังสีอัลตราไวโอเลต ซึ่งจะยังอยู่ในช่วงของแสงที่มองไม่เห็น แต่เมื่อรังสีนี้กระทบสารเรืองแสงที่ฉาบไว้ที่ผิวหลอด สารเรืองแสงจะเปล่งแสงสีต่าง ๆ ตามชนิดของสารเรืองแสงที่ฉาบไว้ในหลอดนั้น ดังที่แสดงในรูปที่ 2.13 ซึ่งโดยทั่วไปหลอดฟลูออเรสเซนต์จะใช้งานร่วมกันกับบัลลาสต์ และสตาร์ทเตอร์ ดังที่แสดงในรูปที่ 2.14 แต่หลอดฟลูออเรสเซนต์ที่ใช้งานร่วมกับบัลลาสต์อิเล็กทรอนิกส์นั้น ไม่ต้องทำงานร่วมกับสตาร์ทเตอร์ ดังแสดงในรูปที่ 2.15

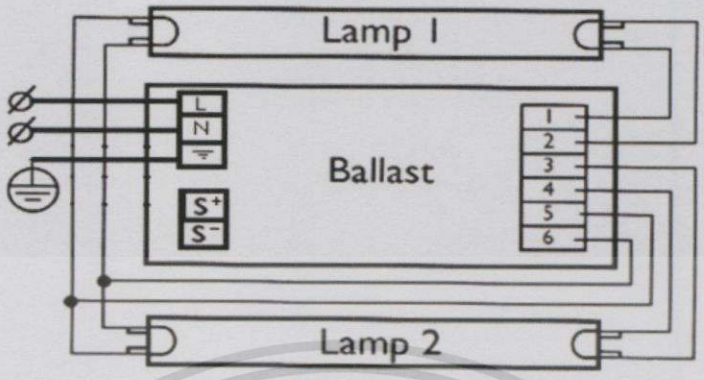


รูปที่ 2.13 ภายในของหลอดฟลูออเรสเซนต์



รูปที่ 2.14 การต่อหลอดฟลูออเรสเซนต์เข้ากับบัลลาสต์และสตาร์ทเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.15 การต่อหลอดฟลูออเรสเซนต์เข้ากับบัลลาสต์อิเล็กทรอนิกส์ แบบ 2 หลอด

2.12 บัลลาสต์อิเล็กทรอนิกส์ (Electronic Ballast)

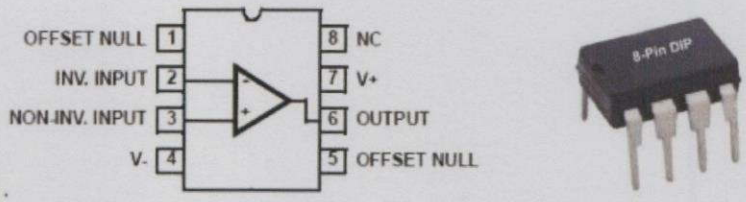
บัลลาสต์อิเล็กทรอนิกส์มีหน้าที่ควบคุมกระแสไฟฟ้าที่ผ่านเข้าไปในขั้วหลอด ให้มีค่าสม่ำเสมอตามแต่ละประเภทหลอด ซึ่งบัลลาสต์อิเล็กทรอนิกส์จะทำหน้าที่เหมือนกับอุปกรณ์ 3 อย่าง เมื่อเทียบกับการใช้งานบัลลาสต์ชนิดขดลวด คือ บัลลาสต์ชนิดขดลวด สตาร์ทเตอร์ และตัวเก็บประจุ โดยที่บัลลาสต์อิเล็กทรอนิคส์จะสร้างความถี่สูงถึง 50 กิโลเฮิรตซ์ (kHz) จ่ายให้กับหลอดฟลูออเรสเซนต์ ซึ่งโดยปกติหลอดฟลูออเรสเซนต์จะตอบสนองต่อความถี่สูง ๆ ได้ดีกว่าความถี่ 50 เฮิรตซ์ ที่ใช้อยู่ในบ้าน เป็นผลให้สามารถประหยัดพลังงานไฟฟ้าได้ไม่น้อยกว่า 30% และยืดอายุการใช้งานของหลอดได้นานกว่า 25 เปอร์เซ็น เมื่อเทียบกับการต่อวงจรโดยใช้บัลลาสต์ชนิดขดลวด การใช้บัลลาสต์อิเล็กทรอนิกส์ยังมีข้อดีอื่น ๆ อีกเช่น หลอดไฟสามารถเปิดติดได้ทันที แสงสว่างที่ออกมาไม่กระพริบให้รำคาญสายตา อีกทั้งไม่สะสมความร้อนเหมือนบัลลาสต์ชนิดขดลวด และหลอดยังคงติดแม้แรงดันไฟฟ้าตก

โดยบัลลาสต์อิเล็กทรอนิกส์ชนิดปรับค่าได้ จะเป็นบัลลาสต์อิเล็กทรอนิกส์อีกชนิดหนึ่งซึ่งมีความสามารถในการปรับแสงสว่างของหลอดไฟ ด้วยวิธีการต่าง ๆ เช่น การใช้แรงดันในการควบคุมความสว่าง เป็นต้น นิยมใช้กันเพื่อลดความสว่างของแสงให้พอดีกับสายตา

2.13 ออปแอมป์

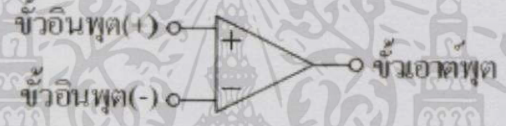
ออปแอมป์หรือวงจรขยายสัญญาณเชิงดำเนินการ (operational amplifier : Op-Amp) ดังรูปที่ 2.16 เป็นอุปกรณ์อิเล็กทรอนิกส์ชนิดหนึ่งที่ภายในประกอบด้วยทรานซิสเตอร์จำนวนหนึ่งทำหน้าที่เป็นวงจรขยายสัญญาณแรงดันด้านอินพุต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

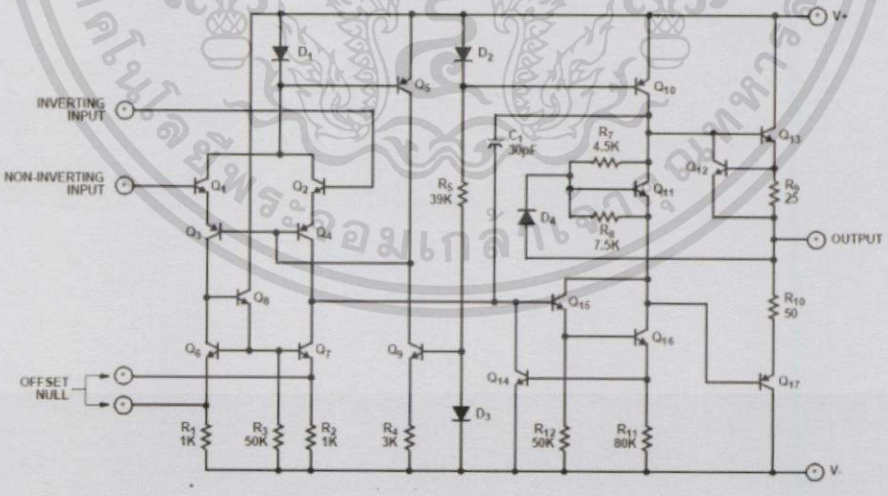


รูปที่ 2.16 ออปแอมป์เบอร์ LM741

โดยสัญลักษณ์ของออปแอมป์จะเป็นรูปสามเหลี่ยม มีขั้วอินพุตสองขั้ว และขั้วเอาท์พุต 1 ขั้ว ดังรูปที่ 2.17 โดยขั้วอินพุตสองขั้วได้แก่ขั้วอินพุตบวก (non-inverting input) และขั้วอินพุตลบ (inverting input)



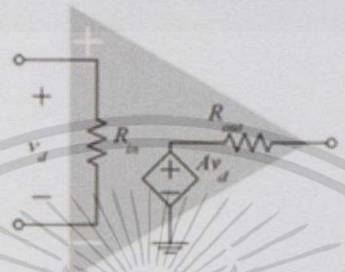
รูปที่ 2.17 สัญลักษณ์ของออปแอมป์



รูปที่ 2.18 วงจรภายในออปแอมป์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วงจรภายในออปแอมป์ประกอบไปด้วยทรานซิสเตอร์จำนวนหนึ่ง แสดงได้ดังในรูปที่ 2.18 โดยสามารถเขียนแทนได้ด้วยวงจรสมมูลอย่างง่ายได้ดังรูปที่ 2.19 โดย R_{in} คือความต้านทานทางด้านอินพุต (input resistance) เป็นความต้านทานที่ต่ออยู่ที่ขั้วอินพุตทั้งสอง R_{out} คือความต้านทานทางด้านเอาต์พุต (output resistance) และ A คืออัตราขยายแรงดันไฟฟ้าวงรอบเปิด (open loop voltage gain)



รูปที่ 2.19 วงจรสมมูลอย่างง่ายของออปแอมป์

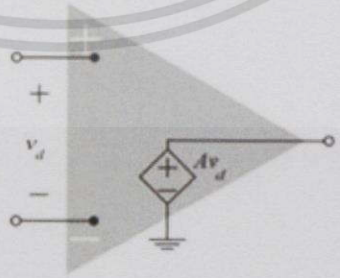
เมื่อกำหนดให้ออปแอมป์เป็นอุดมคติจะได้คุณสมบัติต่าง ๆ ของออปแอมป์ดังนี้

$R_{in} = \infty$ (ความต้านทานทางด้านอินพุตมีค่าเป็นอนันต์)

$R_{out} = 0$ (ความต้านทานทางด้านอินพุตมีค่าเป็นศูนย์)

$A = \infty$ (อัตราขยายแรงดันไฟฟ้าวงรอบเปิดมีค่าเป็นอนันต์)

ดังนั้นจะได้วงจรสมมูลของออปแอมป์อุดมคติดังรูปที่ 2.20 ซึ่งในการวิเคราะห์วงจรที่ประกอบด้วยออปแอมป์นี้ จะใช้วงจรสมมูลของออปแอมป์อุดมคติในการวิเคราะห์ โดยการแทนที่ออปแอมป์ในวงจรด้วยวงจรสมมูลอุดมคติ ในส่วนของค่าอัตราขยายแรงดันไฟฟ้าวงรอบเปิดนั้นถ้าเป็นออปแอมป์อุดมคติจะมีค่าเป็นอนันต์



รูปที่ 2.20 วงจรสมมูลอย่างง่ายของออปแอมป์ในอุดมคติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.14 ไอซี 555

ไอซี 555 หรือ IC 555 Timer ดังรูปที่ 2.21 คือไอซีที่สร้างขึ้นมาเพื่อความสะดวกในการสร้างความถี่ นิยมใช้งานอย่างมากเนื่องจากใช้อุปกรณ์ต่อร่วมน้อย และมีความสเถียร สามารถควบคุมความถี่ได้ด้วยการต่อตัวต้านทาน และ ตัวเก็บประจุ มีขาใช้งานทั้งหมด 8 ขา สามารถทำงานได้ที่แรงดัน 4.5 โวลต์ – 18 โวลต์



รูปที่ 2.21 ไอซี 555

2.14.1 การจัดวงจรใช้งานไอซี 555

สามารถจัดการวงจรได้ 2 แบบ คือ

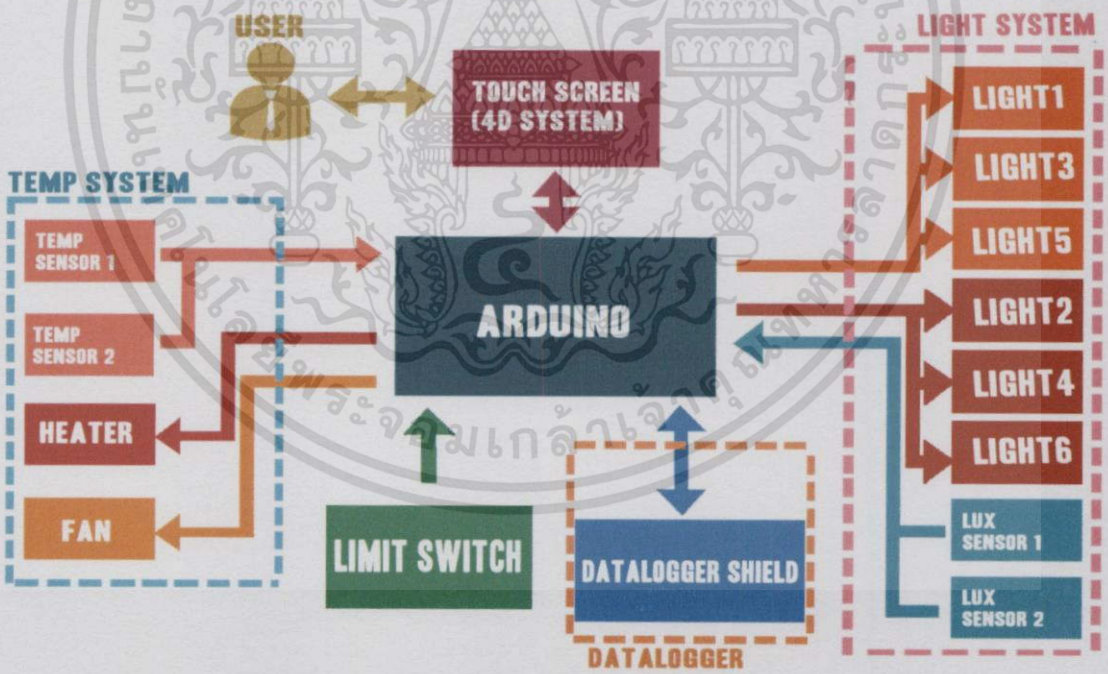
1. โมโนสเตเบิลมัลติไวเบรเตอร์ (Monostable Multivibator) คือ วงจรที่จะสร้างความถี่ขึ้นมา 1 ลูก เมื่อมีการกระตุ้น (Trigger) เกิดขึ้น โดยความกว้างของพัลส์สามารถกำหนดได้ด้วยตัวต้านทานและตัวเก็บประจุ
2. อะสเตเบิลมัลติไวเบรเตอร์ (Astable Multivibator) คือ วงจรสร้างความถี่ขึ้นมาอย่างต่อเนื่อง สามารถกำหนดได้ด้วยตัวต้านทาน (R) จำนวน 2 ตัว และตัวเก็บประจุ (C) 1 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
การออกแบบ

3.1 การออกแบบและการทำงานโดยรวมของระบบ

บทนี้จะกล่าวถึงองค์ประกอบของระบบ การทำงานโดยรวมของระบบ และรายละเอียดในการออกแบบ โดยการออกแบบระบบนั้นมีองค์ประกอบโดยรวมแสดงได้ดังรูปที่ 3.1 ซึ่งในการควบคุมส่วนต่าง ๆ ภายในระบบนั้น จะใช้บอร์ดอาร์ดูโน่เป็นส่วนประมวลผลหลักเพื่อไปควบคุมยังส่วนต่าง ๆ โดยภายในระบบจะประกอบไปด้วย ระบบควบคุมอุณหภูมิ ระบบควบคุมแสงสว่าง ส่วนบันทึกข้อมูล และส่วนแสดงผล โดยระบบควบคุมอุณหภูมินั้นประกอบไปด้วย เซ็นเซอร์วัดอุณหภูมิ ฮีทเตอร์ และพัดลม ส่วนระบบควบคุมแสงสว่างประกอบด้วย เซ็นเซอร์วัดความสว่าง วงจรแปลงแรงดัน บัลลาสต์อิเล็กทรอนิกส์ และหลอดฟลูออเรสเซนต์ ในส่วนของการบันทึกข้อมูลนั้นประกอบด้วย อุปกรณ์บันทึกข้อมูล และ การ์ดหน่วยความจำ และส่วนสุดท้ายคือส่วนการแสดงผลประกอบด้วยหน้าจอระบบสัมผัส

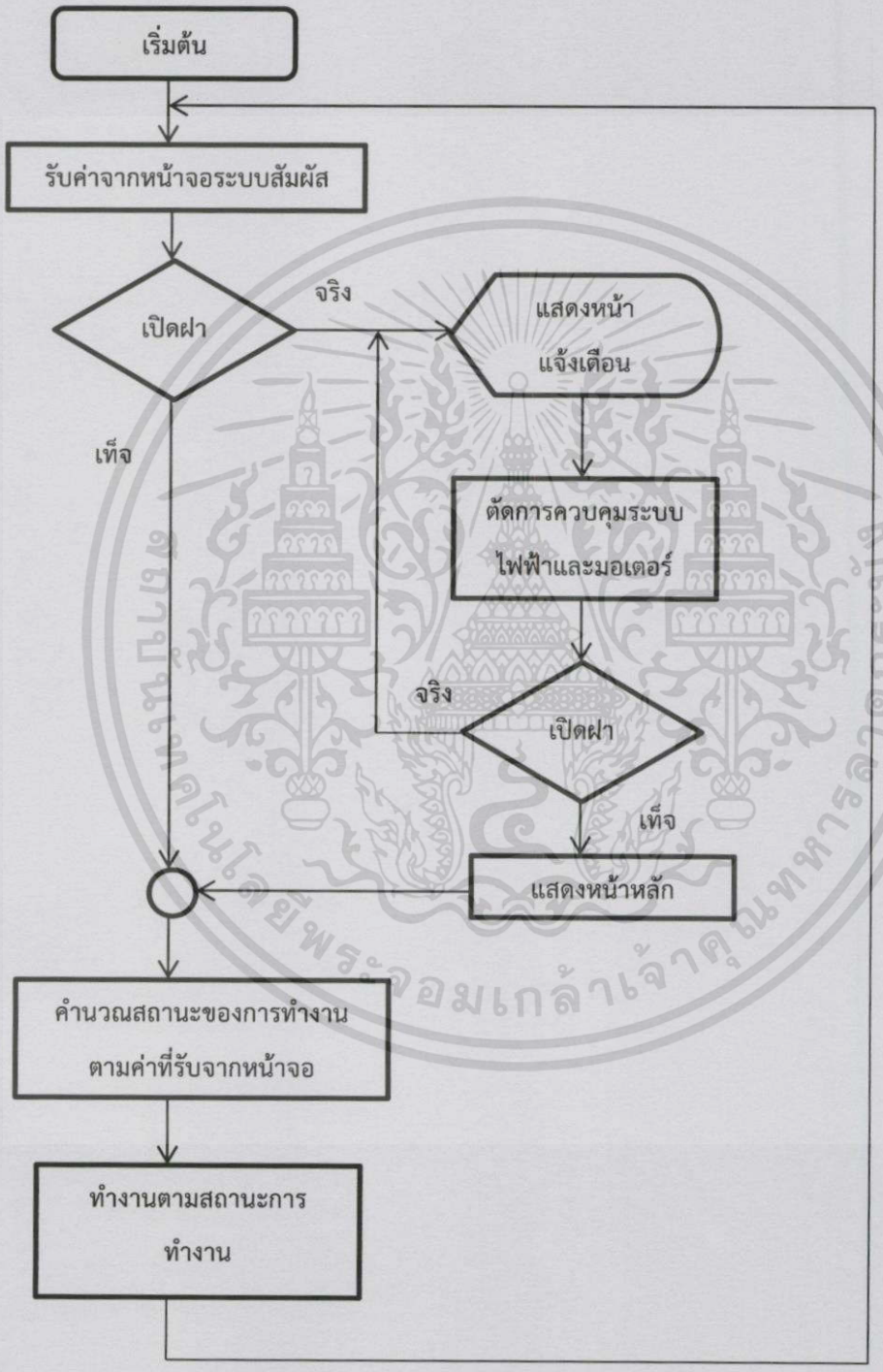


รูปที่ 3.1 บล็อกไดอะแกรมโครงสร้างการทำงานของระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 การออกแบบแผนผังการทำงานของระบบ

3.2.1 โครงสร้างการทำงานหลักของระบบ



รูปที่ 3.2 โครงสร้างการทำงานหลักของระบบ

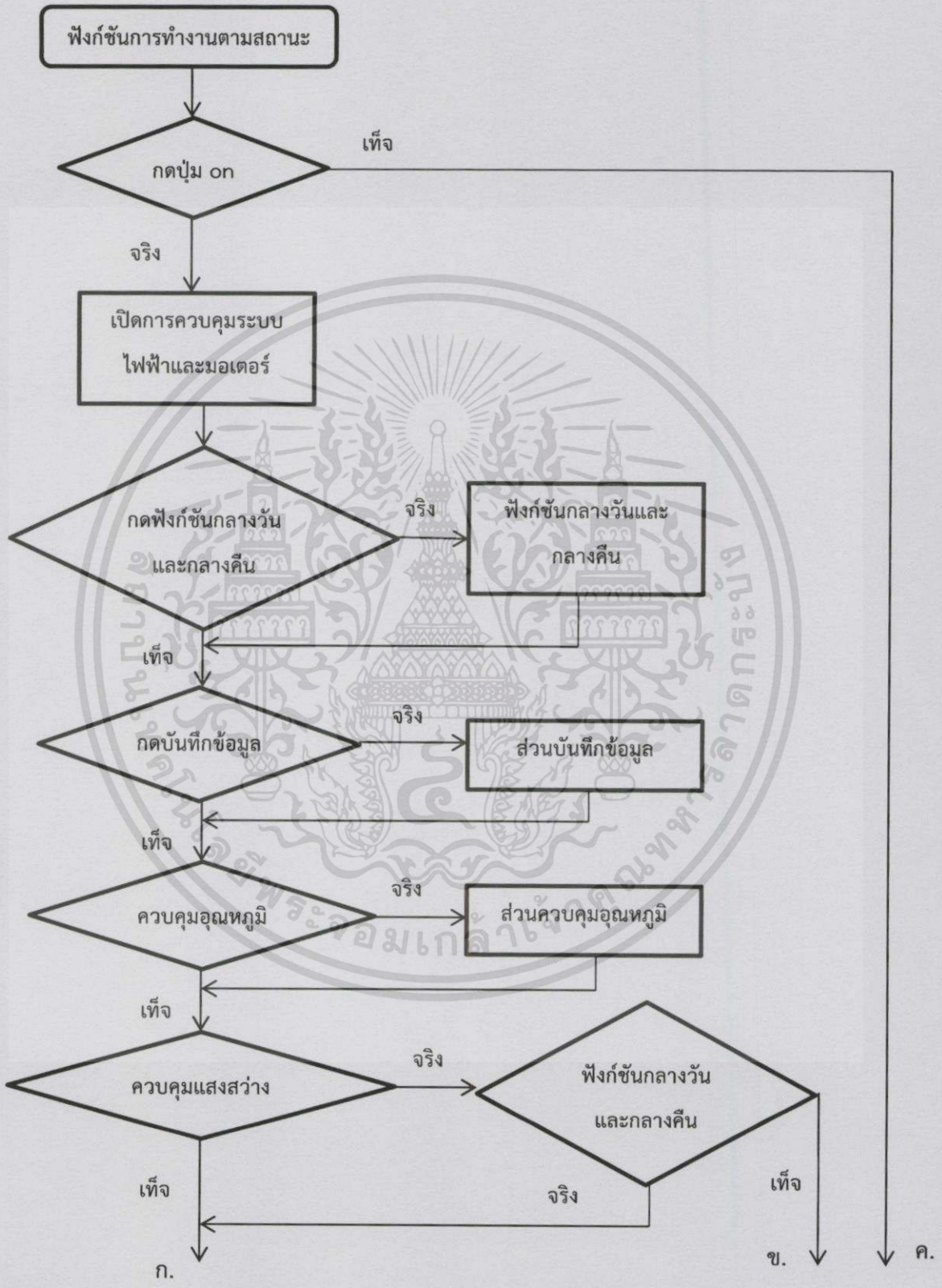
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 3.2 แสดงโครงสร้างการทำงานหลักของระบบ สามารถอธิบายได้ดังนี้ เริ่มต้นด้วยการรับค่าจากหน้าจอระบบสัมผัส หลังจากนั้นจะเข้าสู่การตรวจสอบสถานะของฝาเครื่อง โดยหากฝาเครื่องถูกปิด จะไปยังฟังก์ชันการคำนวณสถานะของการทำงานตามค่าที่ได้รับจากหน้าจอและจะทำงานตามสถานะการทำงานที่คำนวณได้ต่อไป แต่หากฝาเครื่องถูกเปิดอยู่จะแสดงหน้าแจ้งเตือนไปยังผู้ใช้และตัดการควบคุมระบบไฟฟ้าและมอเตอร์ หลังจากนั้นจะตรวจสอบสถานะของฝาอีกครั้ง หากฝาเครื่องยังคงเปิดอยู่ ก็จะวนกลับไปแสดงหน้าจอแจ้งเตือนจนกว่าฝาเครื่องจะถูกปิด จึงจะเข้าสู่ฟังก์ชันคำนวณสถานะการทำงาน และจะวนลูปกลับไปรับค่าจากหน้าจอระบบสัมผัสอีกไปเรื่อยๆ

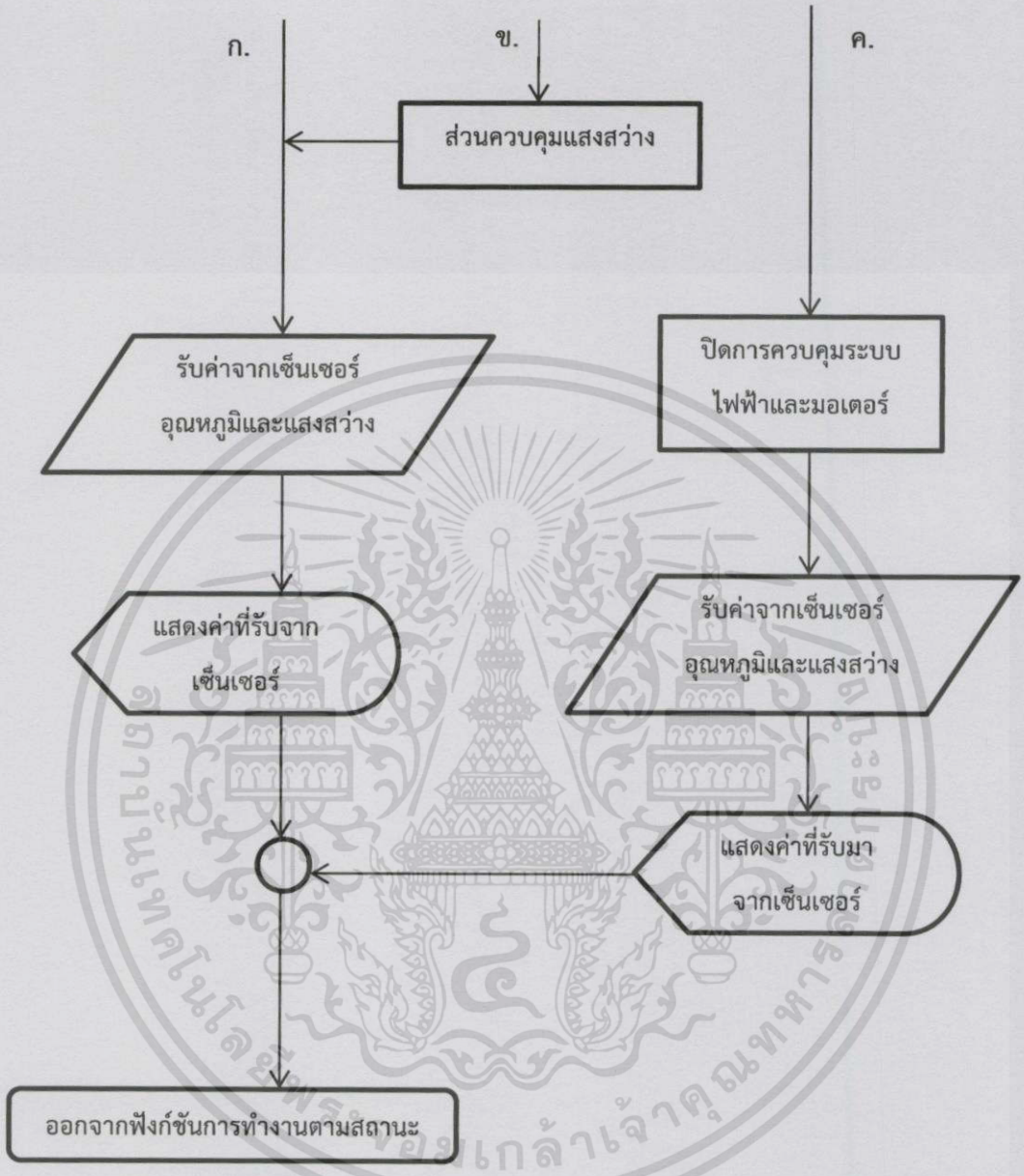
3.2.2 โครงสร้างการทำงานของฟังก์ชันการทำงานตามสถานะ

โครงสร้างการทำงานของฟังก์ชันการทำงานตามสถานะดังรูปที่ 3.3 สามารถอธิบายได้ดังนี้ เมื่อเข้าสู่ฟังก์ชันการทำงานตามสถานะ จะเริ่มต้นด้วยการตรวจสอบปุ่ม on หากไม่มีการกดปุ่ม on จะปิดการควบคุมระบบไฟฟ้าและมอเตอร์ จากนั้นจะรับค่าจากเซ็นเซอร์อุณหภูมิและแสงสว่างไปแสดงยังหน้าจอ และ ออกจากฟังก์ชันการทำงานตามสถานะ แต่หากมีการกดปุ่ม on จะเปิดการควบคุมระบบไฟฟ้าและมอเตอร์ และตรวจสอบการกดปุ่มฟังก์ชันกลางวันและกลางคืน ตรวจสอบการกดปุ่มบันทึกข้อมูล ตรวจสอบการกดปุ่มควบคุมอุณหภูมิ และตรวจสอบการกดปุ่มควบคุมแสงสว่าง หากมีการกดปุ่มใดก็จะเข้าไปทำงานในฟังก์ชันการทำงานในส่วนนั้น ๆ โดยหากมีการกดใช้งานฟังก์ชันกลางวันและกลางคืนอยู่ จะไม่สามารถเข้าไปทำงานในส่วนควบคุมแสงสว่างได้ หลังจากนั้นจะเข้าสู่การรับค่าจากเซ็นเซอร์อุณหภูมิและแสงสว่างไปแสดงยังหน้าจอ และออกจากฟังก์ชันการทำงานตามสถานะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



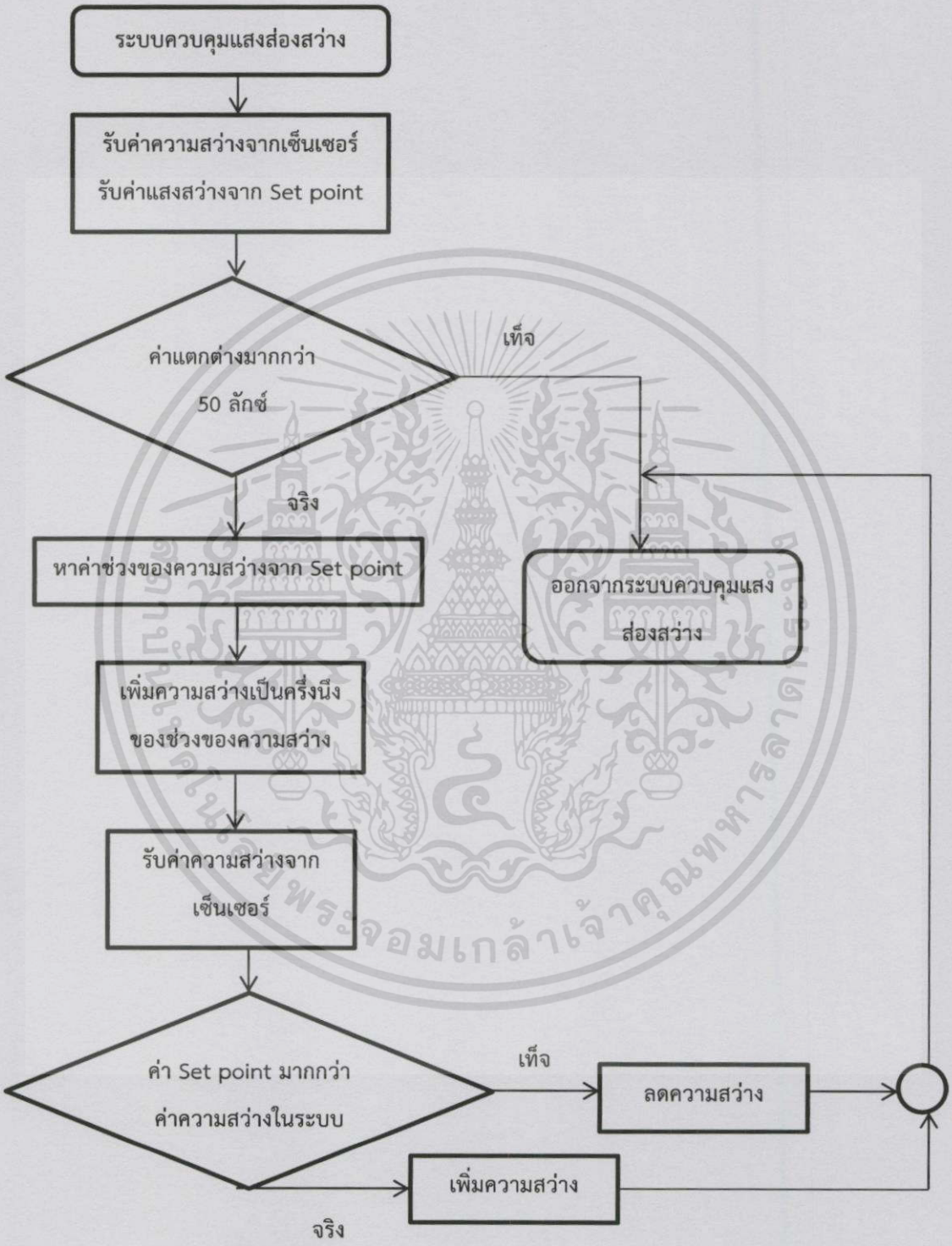
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.3 โครงสร้างการทำงานของฟังก์ชันการทำงานตามสถานะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.3 โครงสร้างการทำงานของฟังก์ชันย่อยระบบควบคุมแสงส่องสว่าง



รูปที่ 3.4 โครงสร้างการทำงานของฟังก์ชันย่อยระบบควบคุมแสงส่องสว่าง

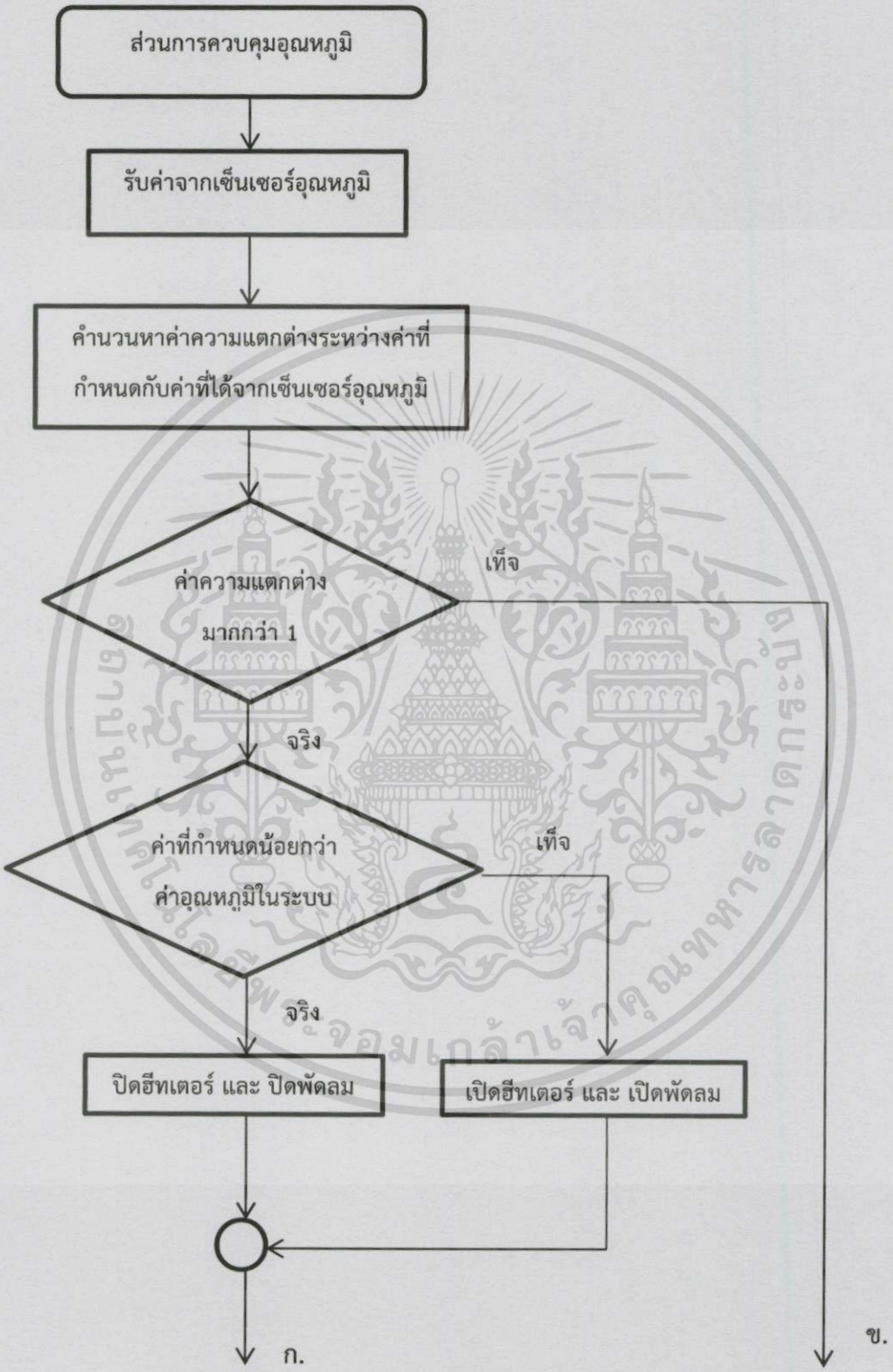
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 3.4 แสดงโครงสร้างการทำงานของฟังก์ชันย่อยระบบควบคุมแสงส่องสว่าง สามารถอธิบายได้ดังนี้ เมื่อเข้าสู่ระบบควบคุมแสงสว่าง จะเริ่มต้นด้วยการรับค่าความสว่างจากเซ็นเซอร์และค่าที่กำหนดไว้ (set point) จากนั้นจึงนำค่าที่ได้มาทำการตรวจสอบ หากมีความแตกต่างไม่มากกว่า 50 ลักซ์ ก็จะออกจากระบบควบคุมอุณหภูมิ แต่หากมีความแตกต่างกันมากกว่า 50 ลักซ์ ก็จะเข้าสู่การหาค่าช่วงของความสว่างจากค่าที่กำหนด แล้วจึงเพิ่มความสว่างให้เท่ากับครึ่งนึงของค่าช่วงที่หาได้ เช่น ค่าที่กำหนดอยู่ในช่วง 4,000-5,000 ลักซ์ ก็จะเพิ่มความสว่างให้เป็น 4,500 ลักซ์ เป็นต้น หลังจากนั้นจะรับค่าจากเซ็นเซอร์แสงอีกครั้ง หากค่าที่กำหนดมากกว่าค่าความสว่างในระบบจริงก็จะทำการเพิ่มค่าความสว่างขึ้น แต่หากค่าที่กำหนดน้อยกว่าค่าความสว่างในระบบก็จะลดค่าความสว่างลง หลังจากนั้นจึงออกจากระบบควบคุมแสงสว่าง

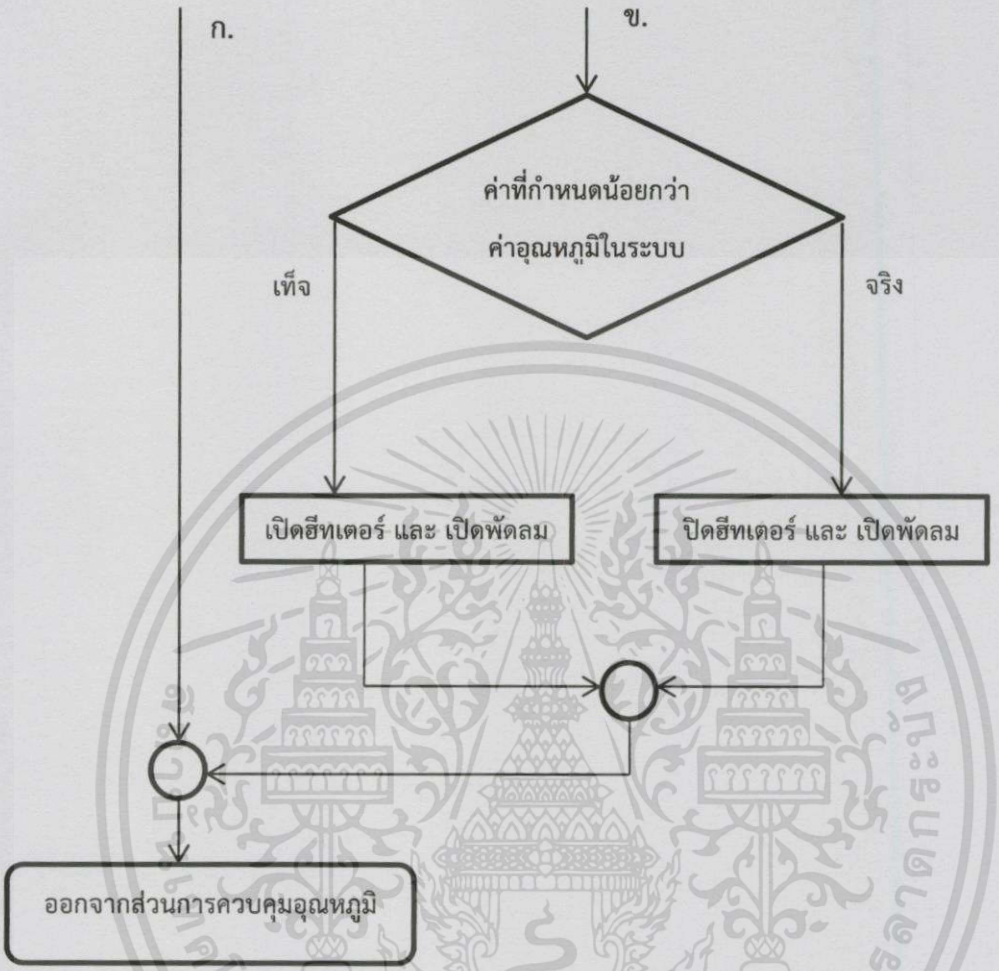
3.2.4 โครงสร้างการทำงานของฟังก์ชันย่อยระบบควบคุมอุณหภูมิ

โครงสร้างการทำงานของฟังก์ชันย่อยระบบควบคุมอุณหภูมิดังรูปที่ 3.5 สามารถอธิบายได้ดังนี้ เมื่อเข้าสู่ระบบควบคุมอุณหภูมิจะทำการรับค่าจากเซ็นเซอร์อุณหภูมิแล้วนำมาคำนวณหาค่าความแตกต่างระหว่างค่าที่กำหนดกับค่าที่ได้รับจากเซ็นเซอร์อุณหภูมิ หากตรวจสอบแล้วค่าที่คำนวณได้แตกต่างกันมากกว่า 1 องศาเซลเซียส จริง ก็จะมาตรวจสอบต่อว่าค่าที่กำหนดนั้นน้อยกว่าอุณหภูมิในระบบหรือไม่ หากน้อยกว่าจริงก็จะทำการปิดฮีทเตอร์และปิดพัดลม แต่หากมากกว่าก็จะทำการเปิดฮีทเตอร์และเปิดพัดลม หลังจากนั้นจึงออกจากระบบควบคุมอุณหภูมิ โดยหากตรวจสอบแล้วค่าที่คำนวณได้ไม่แตกต่างกันมากกว่า 1 องศา ก็จะมาตรวจสอบต่อว่าค่าที่กำหนดน้อยกว่าค่าอุณหภูมิในระบบหรือไม่ หากมากกว่าจริงก็จะทำการปิดฮีทเตอร์และเปิดพัดลม แต่หากไม่มากกว่าก็จำทำการเปิดฮีทเตอร์ละเปิดพัดลม หลังจากนั้นจึงออกจากระบบควบคุมอุณหภูมิ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



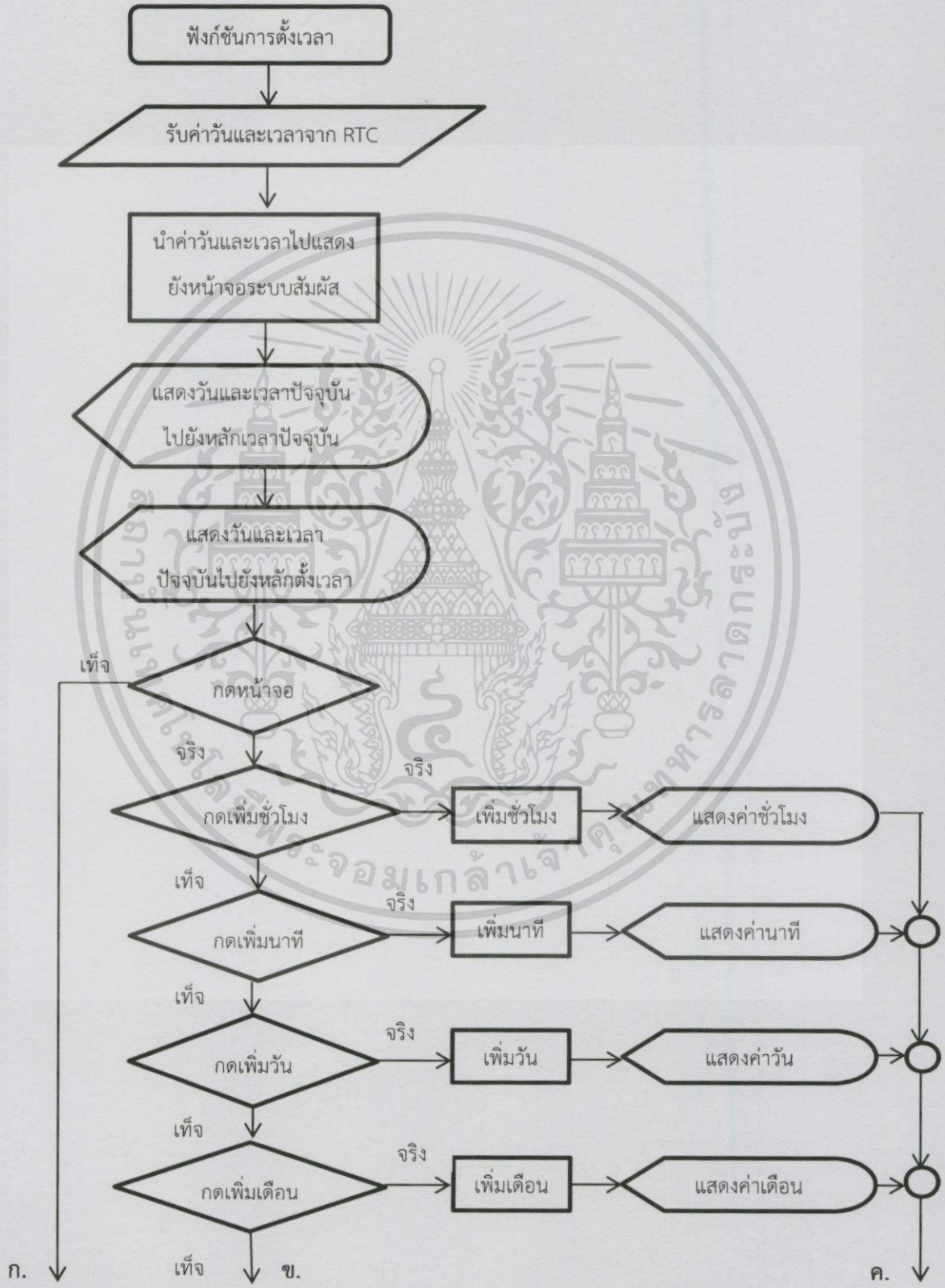
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



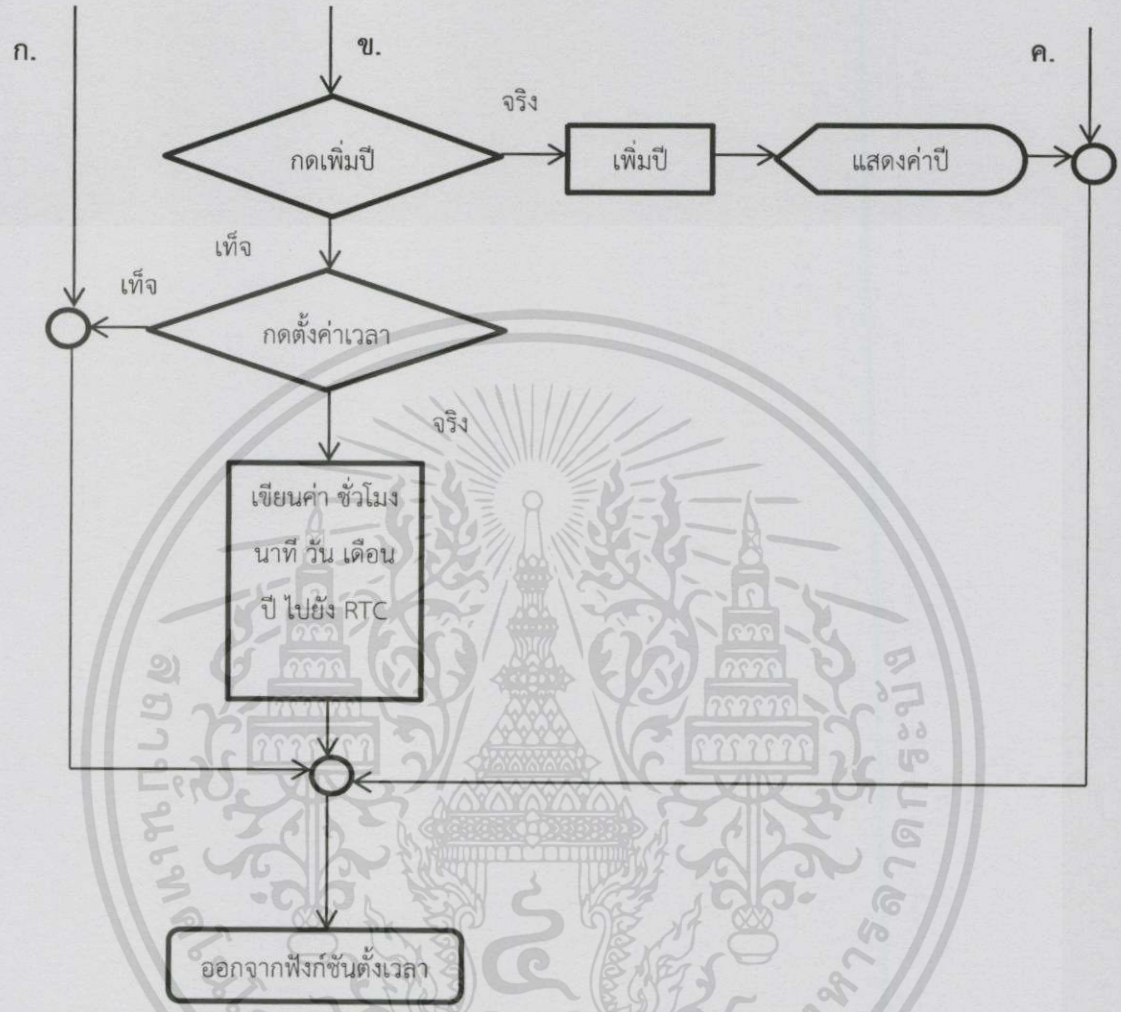
รูปที่ 3.5 โครงสร้างการทำงานของฟังก์ชันย่อยระบบควบคุมอุณหภูมิ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.5 โครงสร้างการทำงานของฟังก์ชันย่อยการตั้งเวลา



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.6 โครงสร้างการทำงานของฟังก์ชันย่อยการตั้งเวลา

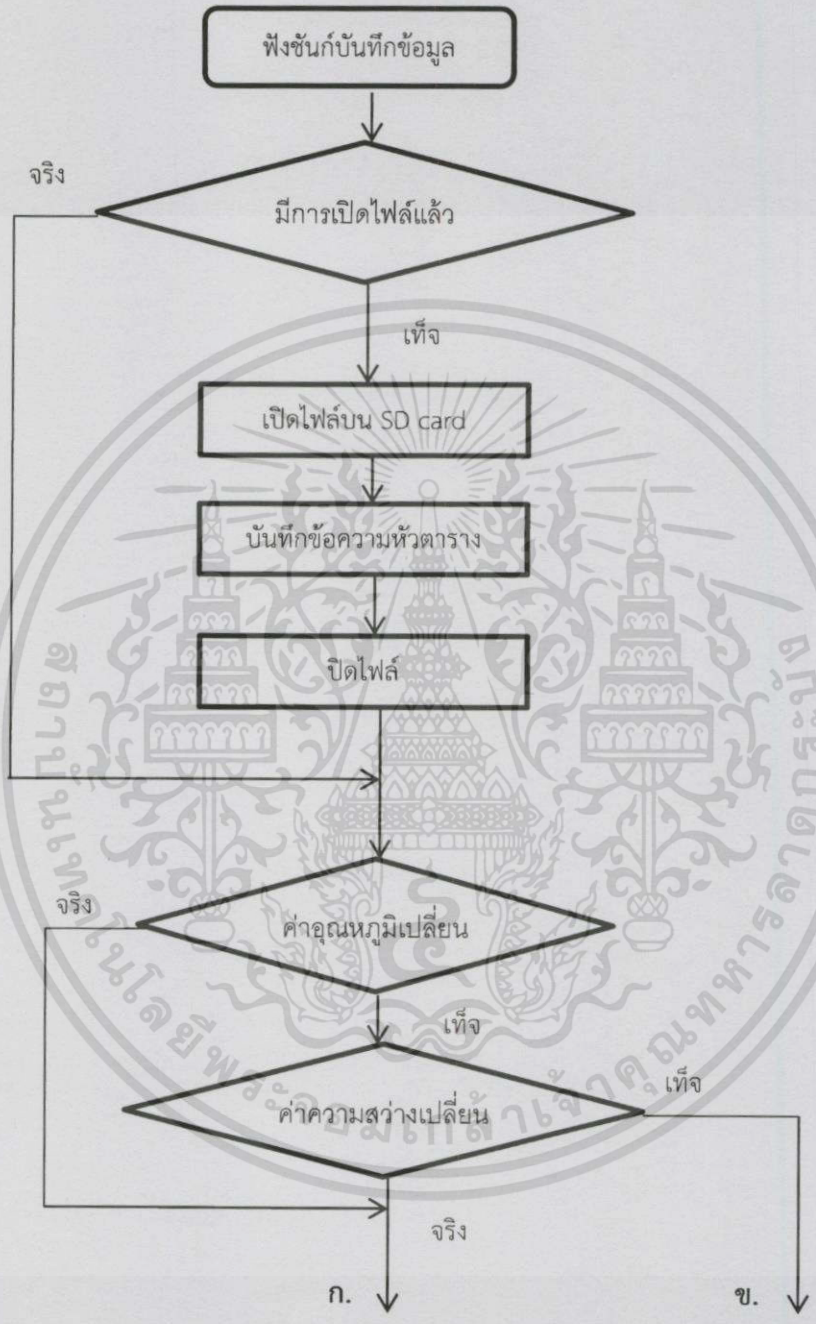
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 3.6 แสดงโครงสร้างการทำงานของฟังก์ชันย่อยการตั้งเวลา สามารถอธิบายได้ดังนี้ เมื่อเข้าสู่ฟังก์ชันการตั้งเวลา จะรับค่าวันและเวลาจากชิพค่าเวลานาฬิกาจริง ไปแสดงยังหน้าจอทั้งในส่วนเวลาปัจจุบัน และ ตั้งเวลา หลังจากนั้นจะทำการตรวจสอบปุ่มหากไม่มีการกดปุ่มตั้งเวลาใด ๆ ก็จะออกจากฟังก์ชันการตั้งเวลา แต่หากมีการกดปุ่มก็จะตรวจสอบว่าปุ่มที่กดนั้นเป็นปุ่มตั้งค่าชั่วโมง นาที วัน เดือน ปี หรือ ปุ่มตั้งค่าเวลา (Set time) หากเป็นปุ่มตั้งค่าชั่วโมงก็จะทำการเพิ่มจำนวนชั่วโมงไปยังหลักชั่วโมง ซึ่งจะทำงานในลักษณะนี้ในทุก ๆ ปุ่ม ที่ใช้ตั้งเวลา แต่หากกดปุ่มตั้งค่าเวลา (Set time) ก็จะทำการนำค่าที่แสดงอยู่ในแต่ละหลัก ไปเขียนลงบนชิพค่าเวลานาฬิกาจริง (RTC) หลังจากนั้นจึงออกจากฟังก์ชันการตั้งเวลา

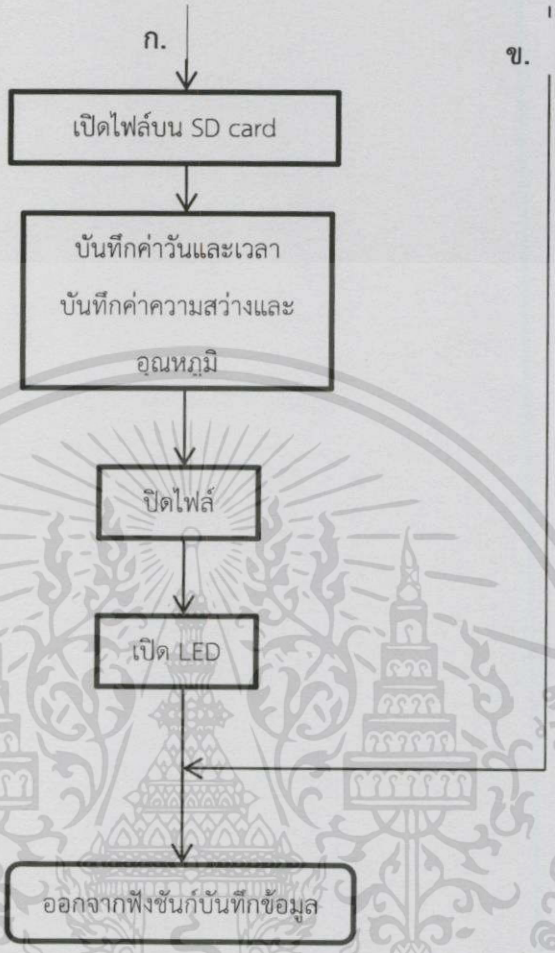
3.2.6 โครงสร้างการทำงานของฟังก์ชันย่อยส่วนบันทึกข้อมูล

โครงสร้างการทำงานของฟังก์ชันย่อยส่วนบันทึกข้อมูลดังแสดงในรูปที่ 3.7 สามารถอธิบายได้ดังนี้ เมื่อเข้าสู่ฟังก์ชันบันทึกข้อมูลจะทำการตรวจสอบว่ามีการเปิดไฟล์ไว้แล้วหรือไม่ หากยังไม่มีการเปิดไฟล์ก็จะทำการเปิดไฟล์ บันทึกหัวตาราง และ ปิดไฟล์ แต่หากมีการเปิดไฟล์แล้ว ก็จะข้ามขั้นตอนการเปิดไฟล์และ เข้าสู่การตรวจสอบค่าอุณหภูมิและค่าความสว่าง หากมีค่าอุณหภูมิหรือค่าความสว่างเปลี่ยนไปค่าใดค่าหนึ่ง ก็จะทำการเปิดไฟล์บนการ์ดหน่วยความจำ เพื่อบันทึกค่าวันและเวลา รวมถึงบันทึกค่าความสว่าง บันทึกค่าอุณหภูมิ และทำการปิดไฟล์ หลังจากนั้นจึงออกจากฟังก์ชันส่วนบันทึกข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



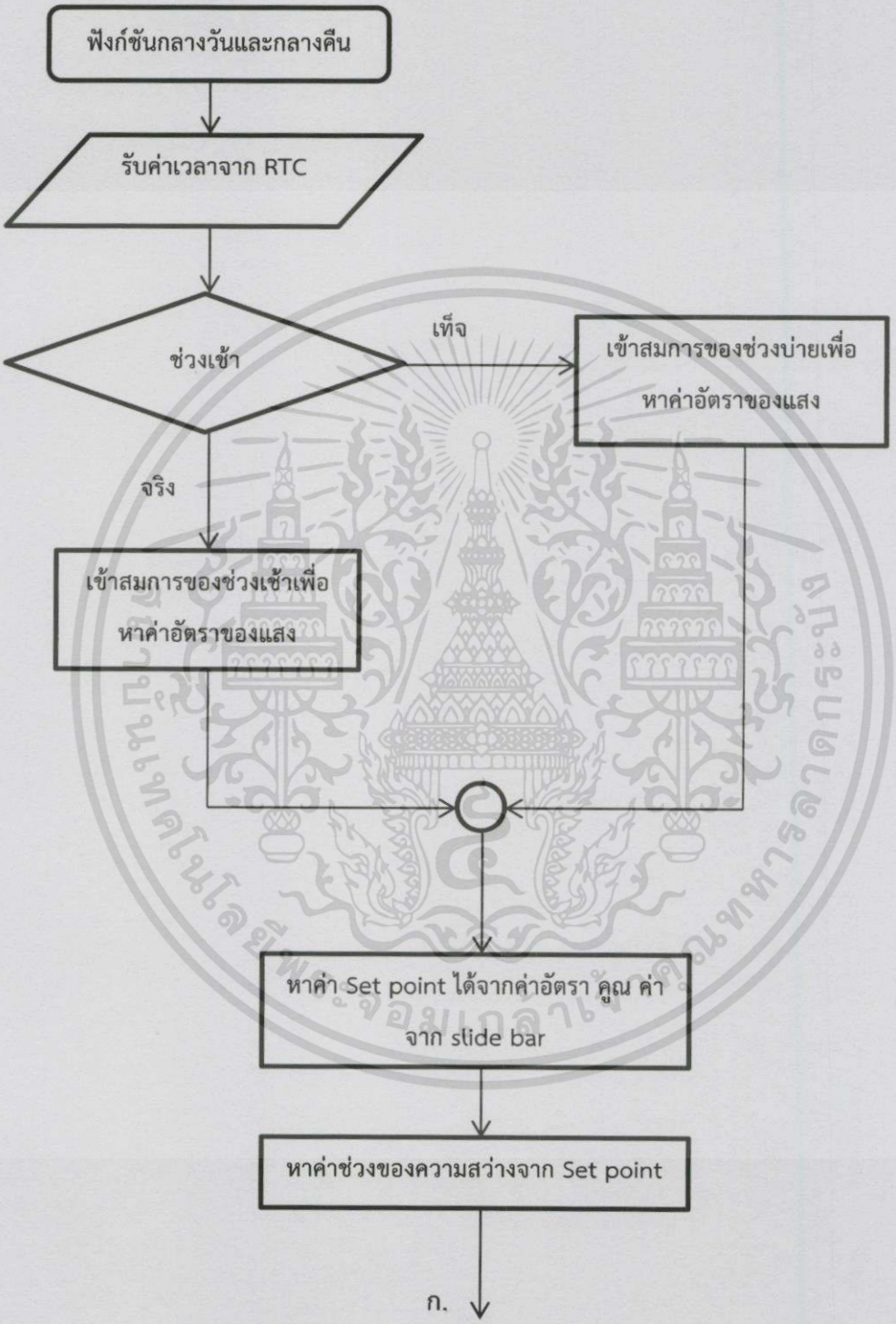
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



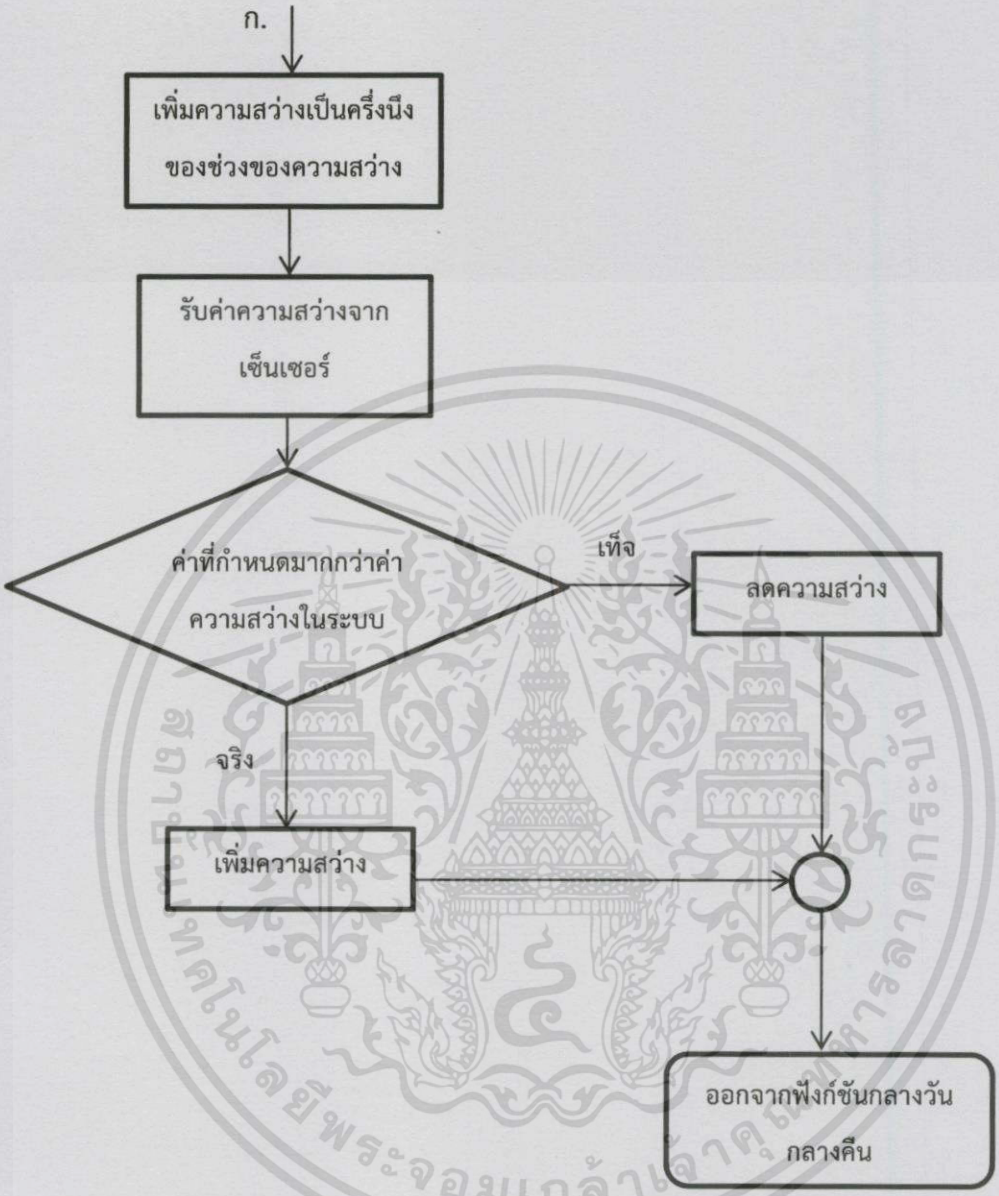
รูปที่ 3.7 โครงสร้างการทำงานของฟังก์ชันย่อยส่วนบันทึกข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.7 โครงสร้างการทำงานของฟังก์ชันย่อยกลางวันและกลางคืน



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.8 โครงสร้างการทำงานของฟังก์ชันย่อยกลางวันและกลางคืน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 3.8 แสดงโครงสร้างการทำงานของฟังก์ชันย่อยกลางวันและกลางคืน สามารถอธิบายได้ดังนี้ เมื่อเข้าสู่ฟังก์ชันกลางวันและกลางคืน ก็จะทำการรับค่าเวลาจากชิพค่าเวลานาฬิกาจริง (RTC) หลังจากนั้นจะทำการตรวจสอบว่าเวลาที่ได้รับนั้นเป็นช่วงเช้าหรือไม่ หากเป็นช่วงเช้าจริง ก็จะเข้าสู่สมการของช่วงเช้าเพื่อหาค่าอัตราของแสง แต่หากไม่ใช่ช่วงเช้าก็จะเข้าสู่สมการของช่วงบ่ายเพื่อหาค่าอัตราของแสง หลังจากนั้นก็จะเข้าสู่การหาค่าที่กำหนด (Set point) ซึ่งได้จากค่าอัตราคูณค่าจากแถบเลื่อน (Slide bar) หลังจากนั้นก็จะเข้าสู่การหาค่าช่วงของความสว่างจากค่าที่กำหนด (Set point) แล้วจึงเพิ่มความสว่างให้เท่ากับครึ่งนึงของค่าช่วงที่หาได้ เช่น ค่าที่กำหนดอยู่ในช่วง 4,000-5,000 ลักซ์ ก็จะเพิ่มความสว่างให้เป็น 4,500 ลักซ์ เป็นต้น หลังจากนั้นจะรับค่าจากเซ็นเซอร์แสง หากค่าที่กำหนดมากกว่าค่าความสว่างในระบบจริงก็จะทำการเพิ่มค่าความสว่างขึ้น แต่หากค่าที่กำหนดน้อยกว่าค่าความสว่างในระบบก็จะลดค่าความสว่างลง หลังจากนั้นจึงออกจากฟังก์ชันกลางวันและกลางคืน

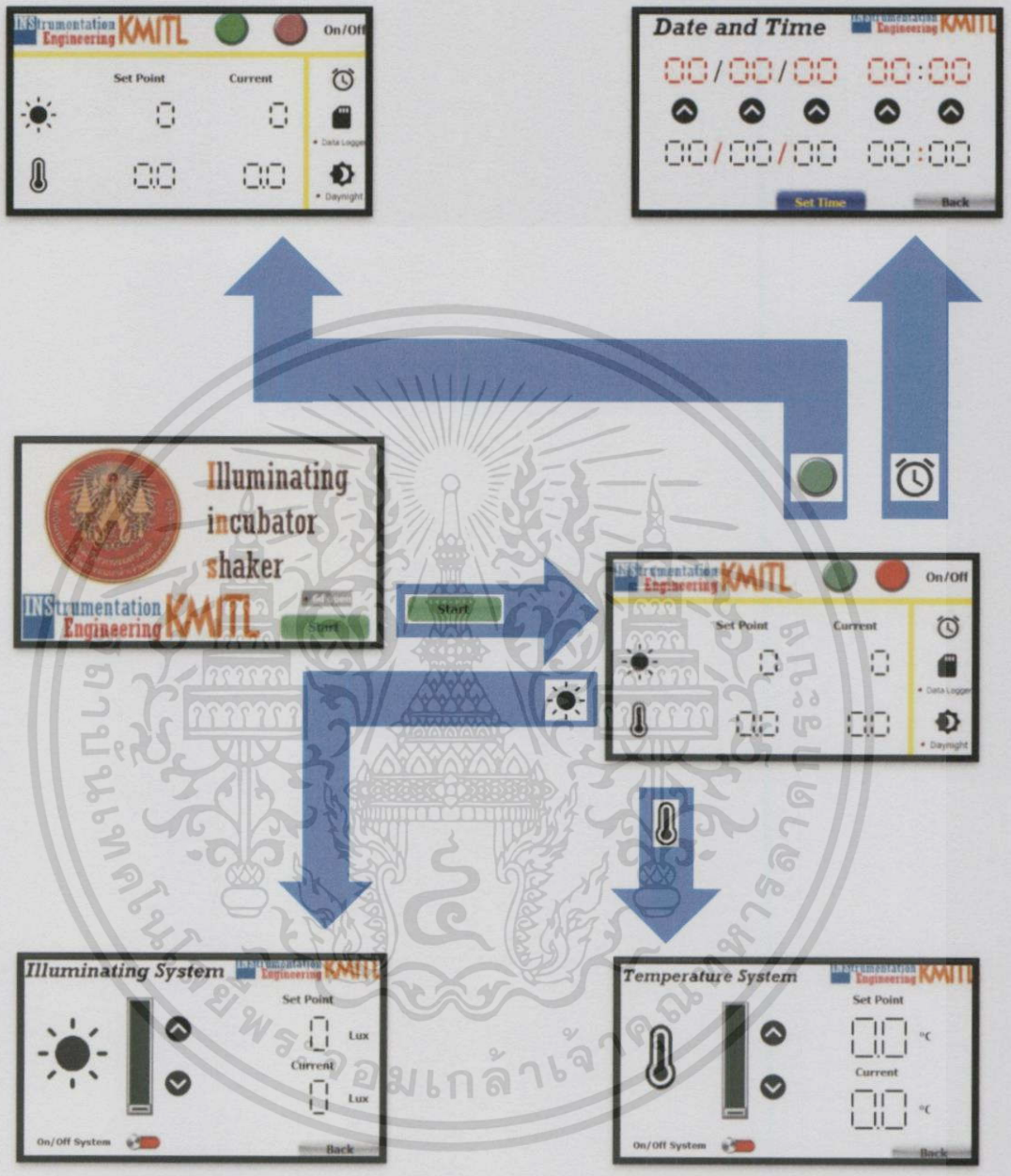
3.3 การออกแบบหน้าจอสำหรับใช้งาน

การออกแบบหน้าจอสำหรับใช้งานนั้น เป็นการออกแบบส่วนแสดงผลบนหน้าจอระบบสัมผัสไปแสดงยังผู้ใช้งาน ผ่านโปรแกรม Workshop4 โดยการออกแบบจะประกอบไปด้วย หน้าจอเริ่มต้น หน้าจอแสดงผลหลัก หน้าจอตั้งค่าอุณหภูมิและแสงสว่าง และหน้าจอตั้งเวลา

3.3.1 โครงสร้างการเชื่อมโยงของหน้าจอระบบสัมผัส

เมื่อกดปุ่ม start จากหน้าเริ่มต้นจะเข้าสู่หน้าจอแสดงผลหลัก ซึ่งเมื่อกดปุ่มต่าง ๆ บนหน้าจอแสดงผลหลัก จะนำไปสู่หน้าจออื่น ๆ ดังแสดงดังรูปที่ 3.9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

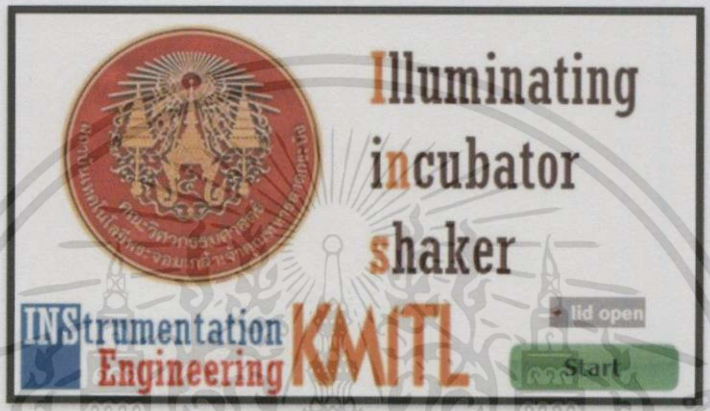


รูปที่ 3.9 การเชื่อมโยงของหน้าจอระบบสัมผัส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.2 หน้าจอเริ่มต้น

สำหรับในหน้าแรกนั้น สามารถแสดงดังรูปที่ 3.10 จะเป็นหน้าที่แสดงเมื่อเครื่องถูกเปิด ซึ่งจะมีไฟแสดงสถานะของฝาเครื่อง ว่ามีสถานะที่เปิดหรือปิดอยู่ หากกดปุ่ม start ในขณะที่ฝาเครื่องถูกเปิดอยู่จะนำไปสู่หน้าเตือนให้ปิดฝาก่อน ดังแสดงในรูปที่ 3.11 แต่หากฝาเครื่องถูกปิดอยู่จะสามารถกดปุ่ม Start เพื่อไปสู่หน้าแสดงผลหลักได้



รูปที่ 3.10 หน้าจอเริ่มต้น

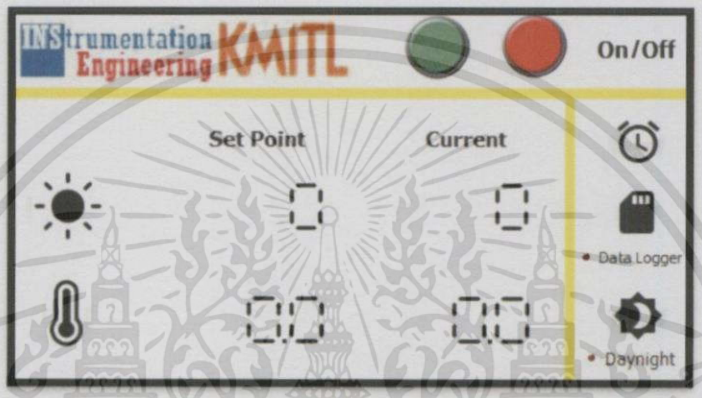


รูปที่ 3.11 หน้าเตือนเมื่อฝาถูกเปิดอยู่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.3 หน้าจอแสดงผล

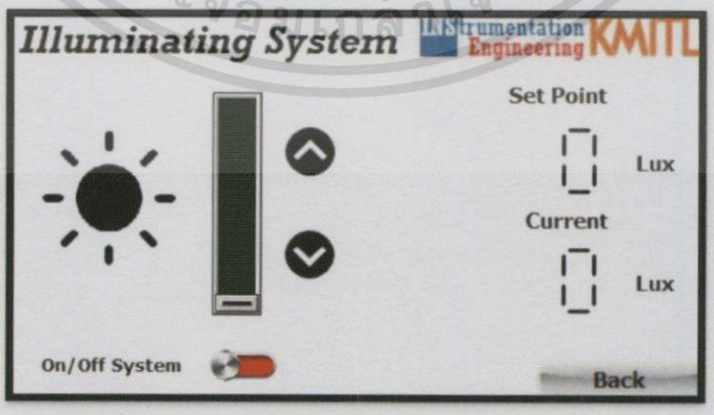
เมื่อกดปุ่ม start จากหน้าจอเริ่มต้น จะเข้ามาในหน้าแสดงผลการทำงานของเครื่องดังรูปที่ 3.12 โดยจะแสดงค่าที่กำหนด (Set point) และค่าที่วัดได้ภายในระบบของค่าความสว่างและอุณหภูมิ (Current) โดยสามารถกดปุ่มสัญลักษณ์รูปไฟและปรอทเพื่อไปยังหน้าการตั้งค่าแสงสว่างและอุณหภูมิ มีปุ่มเลือกโหมดการบันทึกข้อมูลที่เปลี่ยนแปลงภายในระบบ สามารถเลือกโหมดการทำงานจำลองสภาวะแวดล้อมแบบกลางวันและกลางคืน และยังมีปุ่มเพื่อไปยังหน้าตั้งค่าเวลา



รูปที่ 3.12 หน้าจอแสดงผล

3.3.4 หน้าจอตั้งค่าแสงสว่าง

เมื่อมีการกดปุ่มสัญลักษณ์รูปไฟจะนำมาสู่หน้าตั้งค่าแสงสว่างดังแสดงในรูปที่ 3.13 เป็นหน้าที่ใช้ตั้งค่าแสงสว่าง (Set Point) และแสดงค่าแสงสว่างที่วัดได้ภายในระบบ (Current) สามารถเลือกเปิดหรือปิดการควบคุมในส่วนของแสงสว่างได้ และสามารถออกจากหน้าตั้งค่าแสงสว่างกลับไปยังหน้าแสดงผลหลักด้วยการกดปุ่ม back

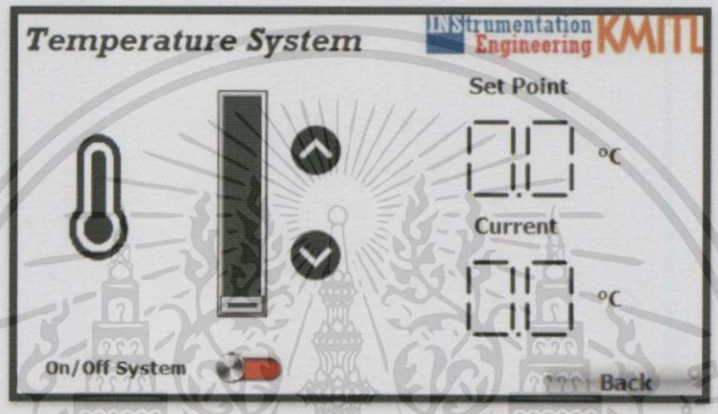


รูปที่ 3.13 หน้าจอตั้งค่าแสงสว่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.5 หน้าจอตั้งค่าอุณหภูมิ

เมื่อกดสัญลักษณ์รูปปรอทจะเข้ามายังหน้า**ตั้งค่า**อุณหภูมิ ดังแสดงในรูปที่ 3.14 เป็นหน้าสำหรับการตั้งค่าอุณหภูมิ (Set point) และแสดงค่าอุณหภูมิที่วัดได้ภายในระบบ (Currrent) สามารถเลือกเปิดหรือปิดในส่วนของการควบคุมอุณหภูมิได้ และสามารถออกจากหน้าตั้งค่าอุณหภูมิกลับไปยังหน้าแสดงผลหลักด้วยการกดปุ่ม back

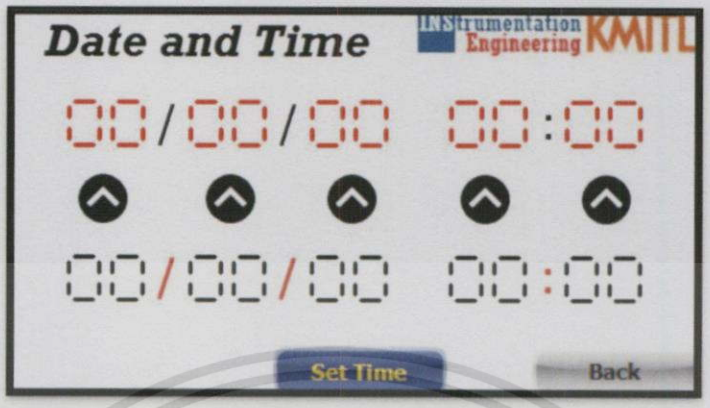


รูปที่ 3.14 หน้าจอตั้งค่าอุณหภูมิ

3.3.6 หน้าจอตั้งค่าเวลา

เมื่อกดสัญลักษณ์รูปนาฬิกาจะเข้ามายังหน้าตั้งค่าเวลา ดังแสดงในรูปที่ 3.15 โดยมีเวลาปัจจุบันที่บันทึกอยู่ในขณะนั้นแสดงในบรรทัดบน ซึ่งสามารถตั้งเวลาของเครื่องให้ตรงกับเวลาตามที่ผู้ใช้ต้องการได้ โดยการตั้งค่า วัน เดือน ปี ชั่วโมง และนาที ด้วยการกดปุ่มขึ้นในแต่ละหลักที่ต้องการเปลี่ยนแปลงค่า ซึ่งค่าที่เปลี่ยนจะมีแต่เพิ่มขึ้นเท่านั้น เมื่อต้องการตั้งค่าให้มีการเปลี่ยนแปลงใหม่ให้กดที่ปุ่ม Set Time และสามารถกลับไปยังหน้าแสดงผลหลักโดยไม่เปลี่ยนแปลงค่าเวลาได้ด้วยการกดปุ่ม back

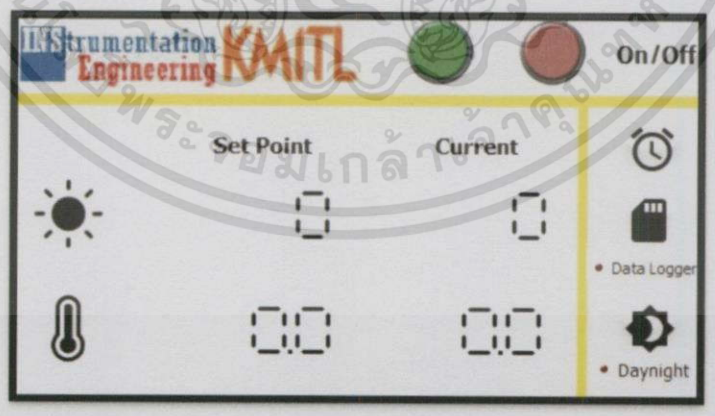
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.15 หน้าจอตั้งค่าเวลา

3.3.7 หน้าจอแสดงสถานะขณะเครื่องกำลังทำงาน

เมื่อมีการกดปุ่ม on หน้าจอจะเปลี่ยนไปสู่หน้าที่แสดงสถานะขณะเครื่องกำลังทำงาน ดังแสดงในรูปที่ 3.16 เนื่องจากได้ทำการออกแบบให้ในขณะที่เครื่องกำลังทำงานอยู่นั้น (หลังจากกดปุ่ม on แล้ว) จะไม่สามารถกดปุ่มอื่นใดเพื่อไปยังหน้าตั้งค่าแสงสว่าง ค่าอุณหภูมิ กดบันทึกข้อมูล และ กดเลือกการทำงานแบบกลางวันและกลางคืนได้ เพื่อเป็นการป้องกันความผิดพลาดจากผู้ใช้งาน ซึ่งอาจจะเกิดจากการไปสัมผัสกับหน้าจอโดยไม่ตั้งใจ ดังนั้นหากต้องการตั้งค่าการทำงานต้องทำการกดปุ่ม off ก่อนเพื่อหยุดการทำงานของเครื่อง หลังจากนั้นจึงจะสามารถเข้าไปตั้งค่าต่าง ๆ ได้



รูปที่ 3.16 หน้าจอแสดงสถานะขณะเครื่องกำลังทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
การดำเนินงาน

4.1 การดำเนินงานและการเชื่อมต่ออุปกรณ์

บทนี้จะกล่าวถึงการดำเนินงานและการเชื่อมต่ออุปกรณ์หลังจากที่ได้ทำการออกแบบ โดยการดำเนินงานจะแบ่งออกเป็นส่วนต่าง ๆ ประกอบด้วย ระบบควบคุมแสงสว่าง ระบบควบคุมอุณหภูมิ ส่วนการตั้งเวลา และส่วนการบันทึกข้อมูล

4.2 ระบบควบคุมแสงสว่าง

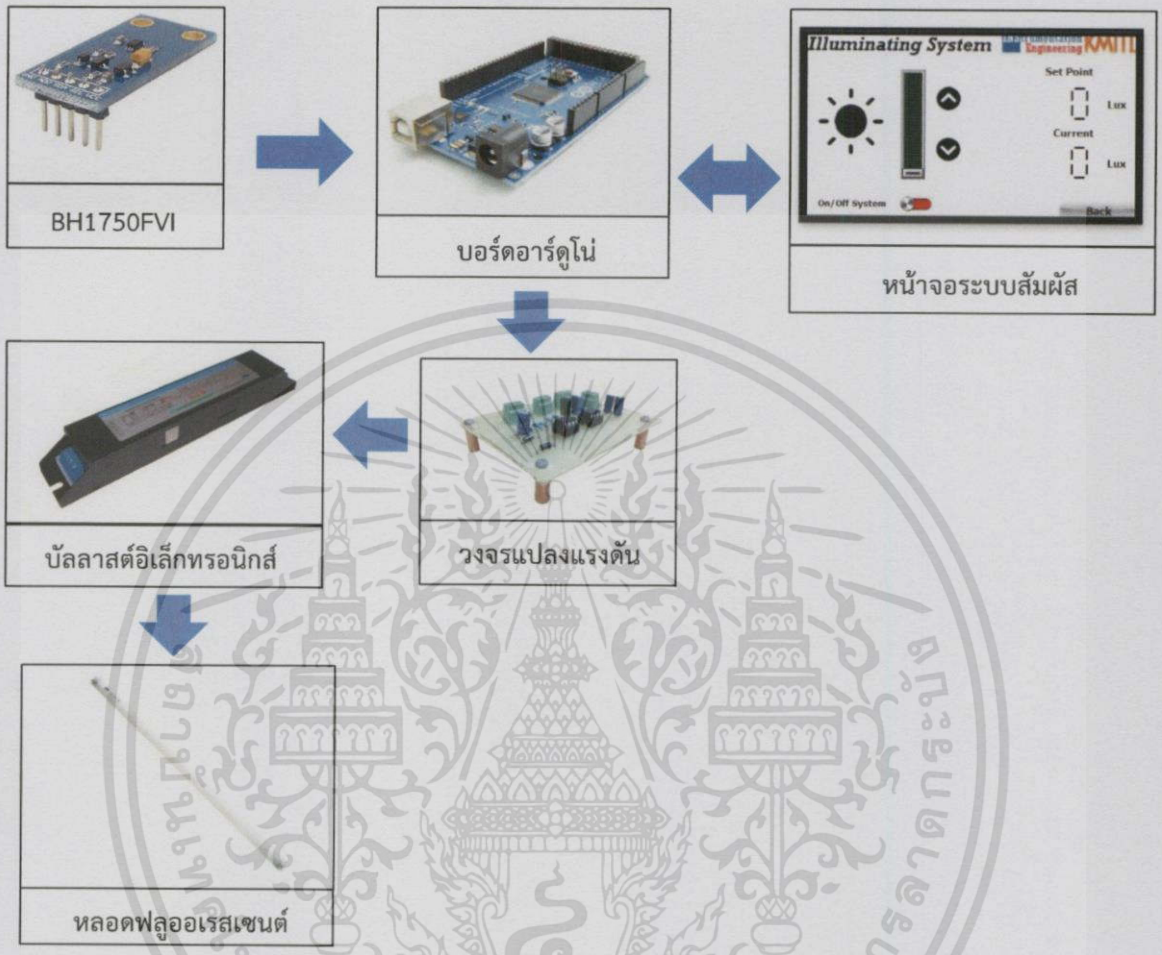
ระบบควบคุมแสงสว่างเป็นระบบที่ใช้ควบคุมแสงสว่างภายในเครื่อง ซึ่งประกอบไปด้วย

1. บอร์ดอาร์ดูโน่ ทำหน้าที่เป็นส่วนควบคุม
2. เซ็นเซอร์แสง (BH1750FVI) ทำหน้าที่ตรวจวัดแสงสว่างภายในเครื่อง
3. วงจรแปลงแรงดัน ทำหน้าที่แปลงแรงดันจากบอร์ดอาร์ดูโน่เพื่อไปควบคุมความสว่างผ่านบัลลาสต์อิเล็กทรอนิกส์
4. บัลลาสต์อิเล็กทรอนิกส์ ทำหน้าที่ปรับความสว่างของหลอดฟลูออเรสเซนต์
5. หลอดฟลูออเรสเซนต์ ทำหน้าที่ให้แสงสว่างภายในเครื่อง
6. หน้าจอระบบสัมผัส ทำหน้าที่เป็นส่วนแสดงผลและกำหนดค่า

4.2.1 โครงสร้างระบบควบคุมแสงสว่าง

จากรูปที่ 4.1 แสดงโครงสร้างระบบควบคุมแสงสว่าง ซึ่งมีบอร์ดอาร์ดูโน่ทำหน้าที่เป็นส่วนควบคุม บอร์ดอาร์ดูโน่จะรับค่าความสว่างจากเซ็นเซอร์แสง และนำมาประมวลผลเพื่อส่งไปควบคุมความสว่างของหลอดฟลูออเรสเซนต์ผ่านวงจรแปลงแรงดันและบัลลาสต์อิเล็กทรอนิกส์ โดยบอร์ดอาร์ดูโน่จะส่งและรับค่าได้จากหน้าจอระบบสัมผัส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

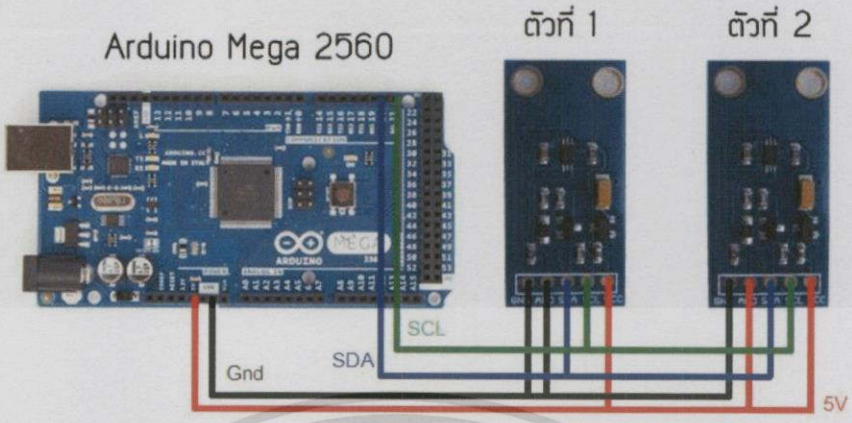


รูป 4.1 โครงสร้างของระบบควบคุมแสงสว่าง

4.2.2 การรับค่าความสว่างจากเซ็นเซอร์วัดแสง (BH1750FVI)

ในส่วนการตรวจวัดค่าความสว่างนั้นจะใช้เซ็นเซอร์ BH1750FVI โมดูล GY-30 จำนวน 2 ตัว ในการตรวจวัดค่าความสว่าง ซึ่งการเชื่อมต่อเป็นไปตามรูปที่ 4.2 โดยสัญญาณที่ได้จากเซ็นเซอร์นั้นเป็นสัญญาณดิจิตอล และสื่อสารกับอาร์ดูโน่แบบ I^2C ผ่านขา SCL และ SDA

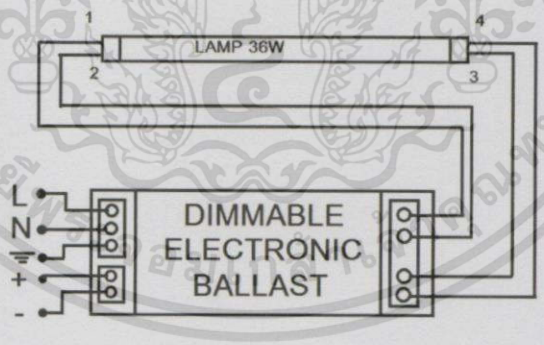
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.2 การเชื่อมต่อ BH1750FVI เข้ากับบอร์ดอาร์ดูโน่

4.2.3 การใช้งานบัลลาสต์อิเล็กทรอนิกส์กับหลอดฟลูออเรสเซนต์

บัลลาสต์อิเล็กทรอนิกส์ชนิดปรับค่าได้นั้น สามารถควบคุมแสงสว่างของหลอดไฟฟลูออเรสเซนต์ โดยการควบคุมผ่านแรงดันไฟฟ้ากระแสตรง 1-10 โวลต์ เข้ามาควบคุมความสว่าง การเชื่อมต่อกับหลอดฟลูออเรสเซนต์เป็นไปตามรูปที่ 4.3 ซึ่งความสว่างจะแปรผันตรงกับแรงดันที่ป้อนให้กับบัลลาสต์อิเล็กทรอนิกส์ชนิดปรับค่าได้

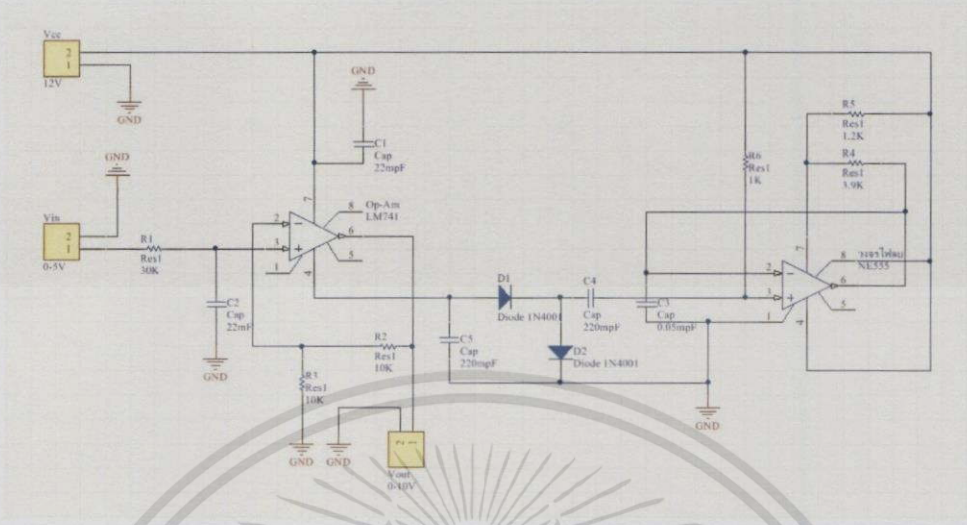


รูปที่ 4.3 การเชื่อมต่อบัลลาสต์อิเล็กทรอนิกส์กับหลอดฟลูออเรสเซนต์

4.2.4 วงจรแปลงแรงดัน

ค่าแรงดันควบคุมที่ถูกส่งมาจากอาร์ดูโน่นั้นอยู่ในช่วง 0-5 โวลต์ ดังนั้นเพื่อให้แรงดันที่จะส่งไปควบคุมแสงสว่างมีความเหมาะสม จึงต้องใช้วงจรขยายแรงดันเข้ามาช่วยเพิ่มแรงดันจาก 0-5 โวลต์ ให้เป็น 0-10 โวลต์ เพื่อส่งไปควบคุมบัลลาสต์อิเล็กทรอนิกส์ โดยมีวงจรขยายแรงดันดังรูปที่ 4.4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.4 การเชื่อมต่อวงจรแปลงแรงดัน

4.3 ระบบควบคุมอุณหภูมิ

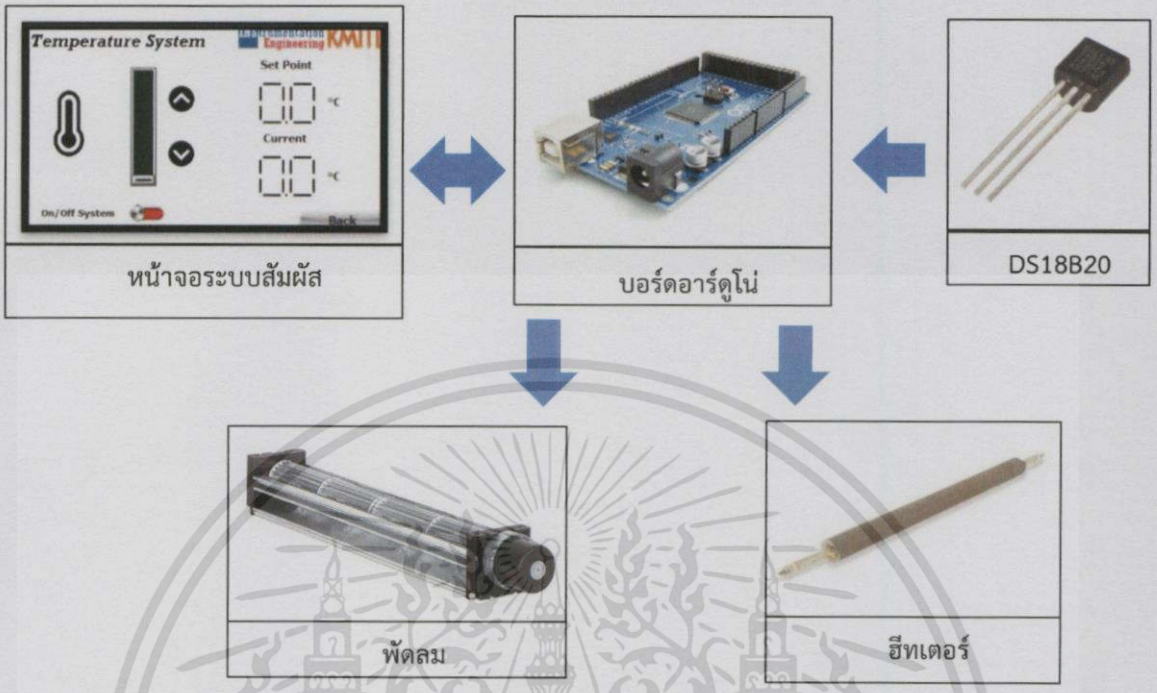
ระบบควบคุมอุณหภูมิเป็นระบบที่ใช้ควบคุมอุณหภูมิภายในเครื่อง ซึ่งประกอบไปด้วย

1. บอร์ดอาร์ดูโน่ ทำหน้าที่เป็นส่วนควบคุม
2. เซ็นเซอร์อุณหภูมิ (DS18B20) ทำหน้าที่ตรวจวัดอุณหภูมิภายในเครื่อง
3. พัดลม ทำหน้าที่นำพาความร้อนจากฮีทเตอร์เข้าสู่ระบบ
4. ฮีทเตอร์ ทำหน้าที่สร้างความร้อน

4.3.1 โครงสร้างของระบบควบคุมอุณหภูมิ

จากรูปที่ 4.5 แสดงโครงสร้างของระบบควบคุมอุณหภูมิ ซึ่งมีบอร์ดอาร์ดูโน่เป็นส่วนควบคุม บอร์ดอาร์ดูโน่จะรับค่าอุณหภูมิจากภายในเครื่อง และมาประมวลผลเพื่อส่งไปควบคุมการเปิดและปิดของพัดลมและฮีทเตอร์ โดยบอร์ดอาร์ดูโน่จะสามารถรับและส่งค่าผ่านทางหน้าจอระบบสัมผัส

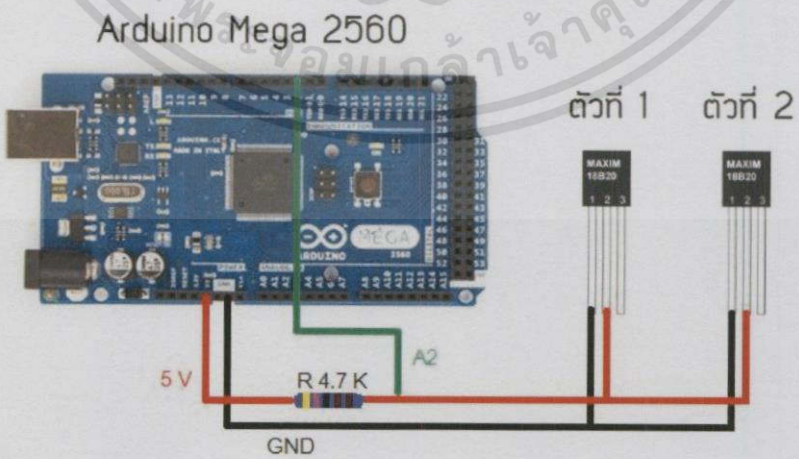
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.5 โครงสร้างของระบบควบคุมอุณหภูมิ

4.3.2 การรับค่าอุณหภูมิจากเซ็นเซอร์อุณหภูมิ (DS18B20)

ในส่วนการตรวจวัดค่าอุณหภูมินั้นจะใช้เซ็นเซอร์ DS18B20 จำนวน 2 ตัว ทำการตรวจวัดอุณหภูมิ โดยสัญญาณที่ได้จากเซ็นเซอร์นั้นเป็นสัญญาณดิจิตอล และสื่อสารกับอาร์ดูโน่ด้วยการสื่อสารแบบสัญญาณ 1 เส้น (1-wire Communication) ซึ่งการเชื่อมต่อเป็นไปตามรูปที่ 4.6



รูปที่ 4.6 การเชื่อมต่อเซ็นเซอร์อุณหภูมิเข้ากับบอร์ดอาร์ดูโน่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.3 การเพิ่มอุณหภูมิภายในระบบด้วยฮีตเตอร์และพัดลม

เมื่อต้องการให้อุณหภูมิในระบบมีอุณหภูมิสูงขึ้นกว่าอุณหภูมิที่วัดได้จากระบบ อาร์ดูโน่จะส่งสัญญาณให้เปิด ฮีทเตอร์ และพัดลม เพื่อทำให้อุณหภูมิในระบบมีค่าสูงขึ้นจนถึงค่าที่กำหนด จึงจะสั่งปิดฮีทเตอร์เพื่อทำให้อุณหภูมิไม่สูงเกินกว่าค่าที่กำหนดไว้ แต่จะยังคงเปิดพัดลมต่อไปอีกชักระยะเพื่อช่วยรักษาระดับความร้อนในตัวเครื่องเขย่า รวมถึงยังเป็นการระบายความร้อนที่สะสมในฮีทเตอร์อีกด้วย แต่หากว่าอุณหภูมิยังสูงกว่าค่าที่กำหนดไว้ก็จะทำการสั่งปิดทั้งฮีทเตอร์และพัดลม

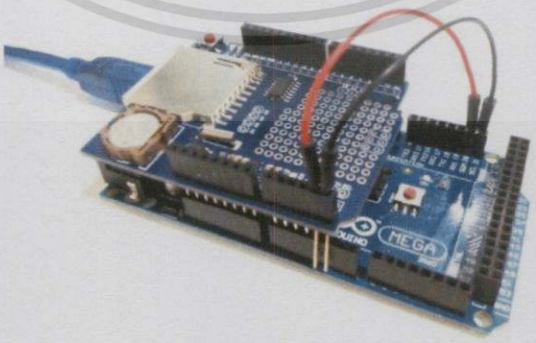
4.4 การตั้งเวลาใช้งาน

ในการรับค่าเวลาปัจจุบันมาใช้งานนั้น จะรับค่าจากชิพค่าเวลานาฬิกาจริง (Real time clock ; DS1307) ซึ่งติดตั้งอยู่บนอุปกรณ์บันทึกข้อมูล โดยการสื่อสารระหว่างอาร์ดูโน่ และ ชิพ DS1307นั้นจะใช้การสื่อสารแบบ I^2C โดยจะเป็นทั้งเส้นทางสำหรับการตั้งค่าเวลาและรับค่าเวลาปัจจุบันจากชิพ

4.4.1 การดัดแปลงชิพค่าเวลานาฬิกาจริงเพื่อใช้งานกับบอร์ดอาร์ดูโน่เมก้า

ดาต้าล้อกเกอร์ชิลล์ที่นำมาใช้งานนั้น ได้ถูกออกแบบมาสำหรับใช้งานบนบอร์ดอาร์ดูโน่ยูโน่เป็นหลัก ซึ่งบอร์ดอาร์ดูโน่ยูโน่กับบอร์ดอาร์ดูโน่เมก้ามีตำแหน่งของขาที่แตกต่างกัน ทำให้จึงต้องดัดแปลงชิพค่าเวลานาฬิกาจริงบนอุปกรณ์บันทึกข้อมูลและบอร์ดอาร์ดูโน่เมก้าให้สามารถทำงานร่วมกันได้ ซึ่งทั้งนี้ทางผู้ผลิตก็ได้อธิบายการดัดแปลงเอาไว้เป็นไปตามดังนี้

1. เชื่อมต่อขา SCL หรือขา A5 บนอุปกรณ์บันทึกข้อมูล ไปยังขา 21 บนบอร์ดอาร์ดูโน่เมก้า
2. เชื่อมต่อขา SDA หรือขา A4 บนอุปกรณ์บันทึกข้อมูล ไปยังขา 20 บนบอร์ดของอาร์ดูโน่เมก้า ดังภาพที่ 4.7

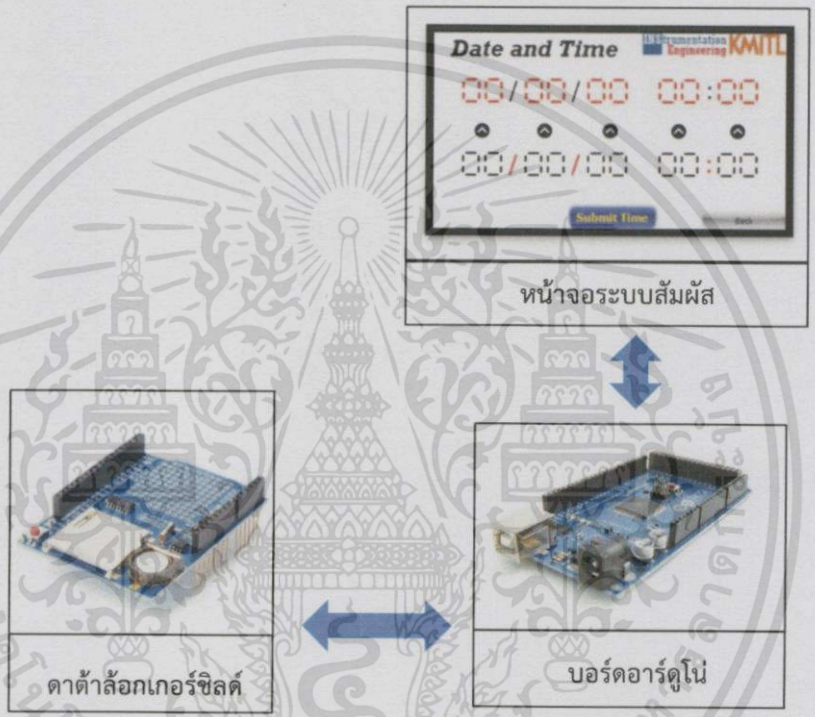


รูปที่ 4.7 การดัดแปลงใช้งานชิพค่าเวลานาฬิกาจริงให้สามารถทำงานร่วมกับบอร์ดอาร์ดูโน่เมก้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4.2 โครงสร้างการแสดงผลและตั้งเวลาผ่านหน้าจอระบบสัมผัส

จากรูปที่ 4.8 แสดงโครงสร้างการแสดงผลและตั้งเวลาผ่านหน้าจอระบบสัมผัส ซึ่งมีบอร์ดอาร์ดูโน่ทำหน้าที่รับและส่งข้อมูลระหว่างหน้าจอระบบสัมผัส และชิพค่าเวลานาฬิกาจริง บอร์ดอาร์ดูโน่จะรับค่าเวลาปัจจุบันจากชิพค่าเวลานาฬิกาจริง (RTC) แล้วส่งเวลาปัจจุบันนั้นไปแสดงยังหน้าจอระบบสัมผัส ในขณะเดียวกันบอร์ดอาร์ดูโน่ก็จะรับค่าเวลาที่ตั้งจากหน้าจอระบบสัมผัสไปเขียนลงบนชิพค่าเวลานาฬิกาจริง ทำให้ค่าที่รับจากชิพค่าเวลานาฬิกาจริงเปลี่ยนไปตามเวลาที่ตั้ง



รูปที่ 4.8 การแสดงผลและตั้งเวลาผ่านหน้าจอระบบสัมผัส

4.5 การบันทึกข้อมูลด้วยอุปกรณ์บันทึกข้อมูล (Datalogger shield)

โดยทั่วไปภายในบอร์ดอาร์ดูโน่นั้นจะมีหน่วยความจำภายในตัวเอง แต่มีขนาดเพียงไม่กี่ร้อยไบต์เท่านั้น จึงทำให้ไม่สามารถเก็บข้อมูลในระยะยาวได้ ในขณะที่การ์ดหน่วยความจำมีขนาดหน่วยความจำที่มากกว่าในระดับกิ๊กกะไบต์ ทำให้สามารถเก็บข้อมูลได้มาก ซึ่งทำให้มีการออกแบบอุปกรณ์บันทึกข้อมูลขึ้นมาสำหรับใช้ในการบันทึกข้อมูลลงบนการ์ดหน่วยความจำ โดยการสื่อสารระหว่างส่วนบันทึกข้อมูลและบอร์ดอาร์ดูโน่เมก้านั้น จะสื่อสารแบบ SPI โดยบอร์ดอาร์ดูโน่ทำหน้าที่เป็นมาสเตอร์และอุปกรณ์บันทึกข้อมูลทำหน้าที่เป็นสเลฟ ซึ่งจะใช้ 4 เส้นทางในการสื่อสารคือ SCLK MOSI MISO และ SS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.5.1 การดัดแปลงส่วนของการบันทึกข้อมูลให้สามารถนำมาใช้งานร่วมกับบอร์ดอาร์ดูโน่เมก้า

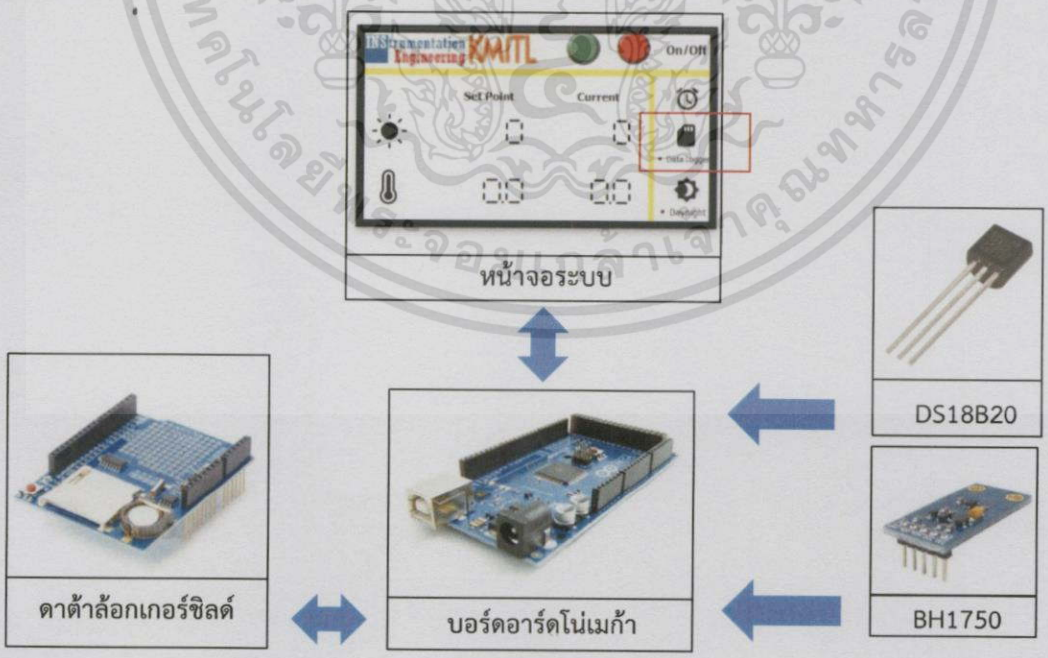
เนื่องด้วยบอร์ดอาร์ดูโน่เมก้ามีขาที่ใช้ในการสื่อสารแบบ SPI ที่ไม่เหมือนกับบอร์ดอาร์ดูโน่ยูโน่ ทำให้ในการตั้งค่าขาต้องมีการระบุขาสำหรับการสื่อสารที่แตกต่างกันด้วย โดยสำหรับบอร์ดอาร์ดูโน่เมก้าให้ใช้ขา 10 11 12 13 โดยมีการตั้งค่าดังรูปที่ 4.9 นอกจากนี้ยังต้องมีการเปลี่ยนไลบรารี่ซึ่งจะเป็น SD ไลบรารี่สำหรับบอร์ดอาร์ดูโน่เมก้าโดยเฉพาะ สามารถดาวน์โหลดได้ผ่าน https://github.com/adafruit/SD (http://adafru.it/aP6)

```
// see if the card is present and can be initialized:
if (!SD.begin(10, 11, 12, 13)) {
```

รูปที่ 4.9 การตั้งค่าขาที่ใช้ในการสื่อสารแบบ SPI ระหว่างส่วนบันทึกข้อมูลกับบอร์ดอาร์ดูโน่เมก้า

4.5.2 โครงสร้างการบันทึกข้อมูลด้วยอุปกรณ์บันทึกข้อมูล

จากรูปที่ 4.10 แสดงโครงสร้างการบันทึกข้อมูลด้วยอุปกรณ์บันทึกข้อมูล ซึ่งมีบอร์ดอาร์ดูโน่ทำหน้าที่ควบคุมการบันทึก เมื่อมีคำสั่งเริ่มการบันทึกข้อมูลจากหน้าจอระบบสัมผัส บอร์ดอาร์ดูโน่จะรับค่าความสว่างและค่าอุณหภูมิจากเซ็นเซอร์มาประมวลผล แล้วส่งไปบันทึกยังอุปกรณ์บันทึกข้อมูล



รูปที่ 4.10 โครงสร้างการบันทึกข้อมูลด้วยอุปกรณ์บันทึกข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
ผลการทดลอง

5.1 บทนำ

ในบทนี้จะกล่าวถึงการทดลอง และผลการทดลองของโครงงานเครื่องเขย่าแบบควบคุมอุณหภูมิและแสงส่องสว่าง โดยจะแบ่งการทดลองออกเป็น 4 ลักษณะคือ การทดลองการทำงานของระบบควบคุมอุณหภูมิ การทดลองการทำงานของระบบควบคุมแสงสว่าง การทดลองการทำงานของระบบควบคุมแสงสว่างในโหมดกลางวันและกลางคืน และการทดลองการทำงานของระบบควบคุมแสงสว่างร่วมกับอุณหภูมิ

5.2 การทดลองและผลการทดลองการทำงานของระบบควบคุมอุณหภูมิ

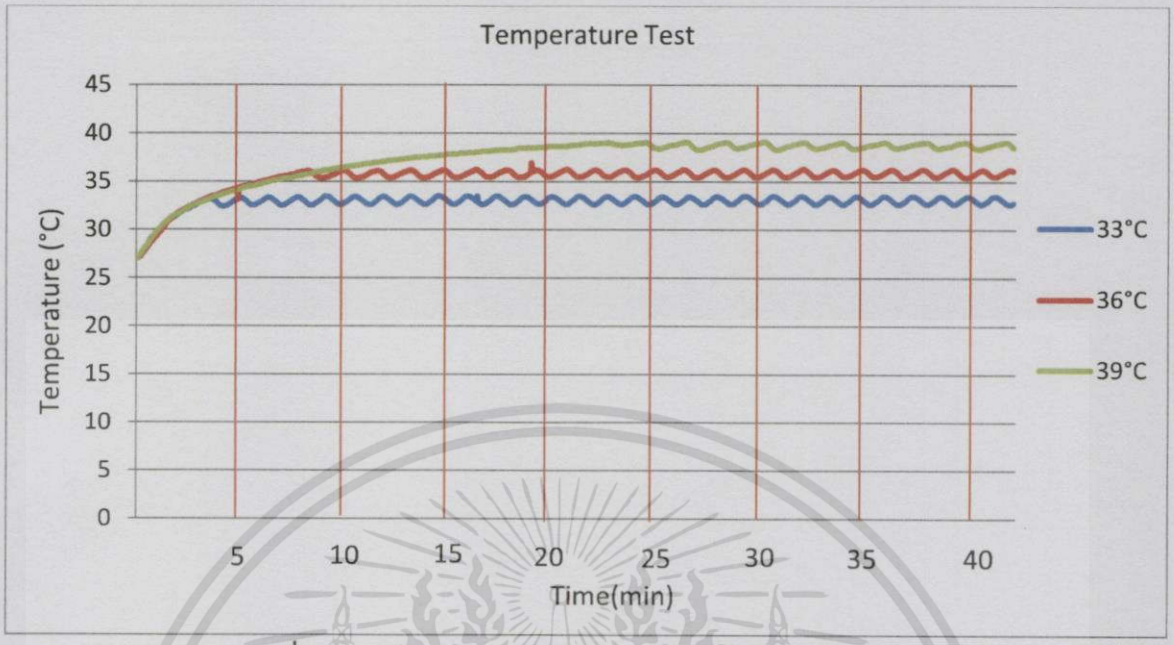
5.2.1 การทดลองการทำงานของระบบควบคุมอุณหภูมิ

การทดลองการทำงานของระบบควบคุมอุณหภูมิ เป็นการทดลองเพื่อทดสอบประสิทธิภาพการทำงานของเครื่องเขย่า ว่ามีความสามารถในการเพิ่มความร้อนให้กับระบบมากน้อยเพียงใด ภายใต้สภาวะแวดล้อมที่มีการควบคุมอุณหภูมิภายนอก กล่าวคือทำการทดลองภายในห้องที่มีการเปิดเครื่องปรับอากาศอยู่ตลอดเวลา ทั้งนี้อุณหภูมิเริ่มต้นจะอยู่ที่ 27 องศาเซลเซียส สำหรับค่าอุณหภูมิที่ใช้สำหรับการทดสอบมีด้วยกัน 3 ค่า คือ 33 36 และ 39 องศาเซลเซียส

5.2.2 ผลการทดลองการทำงานของระบบควบคุมอุณหภูมิ

จากการทดลองพบว่า ตัวเครื่องสามารถควบคุมอุณหภูมิภายในตัวเครื่องให้มีอุณหภูมิเท่ากับค่าที่กำหนดไว้ได้ โดยอุณหภูมิที่มีค่าสูงจะต้องใช้เวลาในการปรับค่าอุณหภูมิให้เท่ากับค่าที่ตั้งไว้มากกว่าอุณหภูมิที่มีค่าต่ำกว่า และหลังจากอุณหภูมิภายในตัวเครื่องมีค่าเท่ากับค่าที่กำหนดไว้ ระบบควบคุมสามารถรักษาระดับอุณหภูมิให้มีค่าใกล้เคียงกับค่าที่กำหนดไว้ได้ แต่จะมีการเปลี่ยนแปลงอยู่ในช่วง 0.5 องศาเซลเซียส ซึ่งถือว่ามีเสถียรภาพอยู่พอสมควร ดังแสดงตามรูปที่ 5.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.1 ผลการทดลองการทำงานของระบบควบคุมอุณหภูมิ

5.3 การทดลองและผลการทดลองการทำงานของระบบควบคุมแสงสว่าง

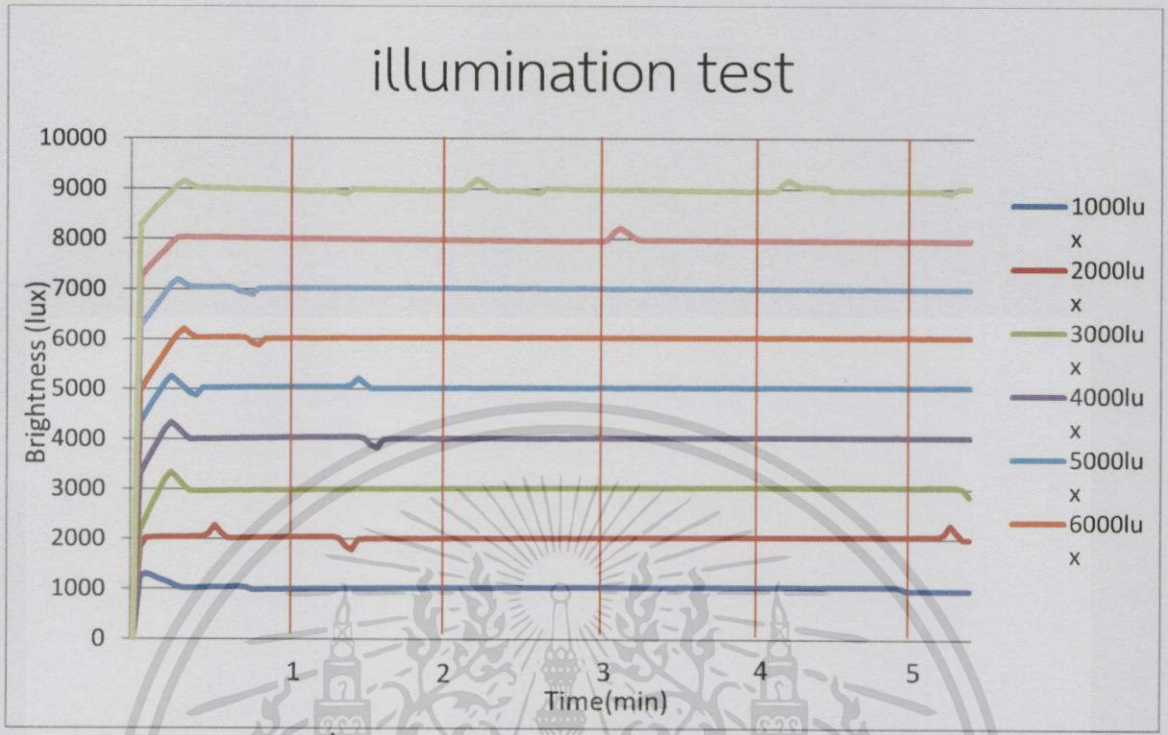
5.3.1 การทดลองการทำงานของระบบควบคุมแสงสว่าง

การทดลองการทำงานของระบบควบคุมแสงสว่าง เป็นการทดลองเพื่อทดสอบประสิทธิภาพในการให้แสงสว่างภายในตัวเครื่องมีค่าเท่ากับค่าแสงสว่างที่กำหนดไว้ และรักษาระดับความสว่างของแสงให้มีค่าใกล้เคียงกับค่าที่กำหนดไว้ การทดลองเริ่มต้นตั้งแต่เริ่มการสั่งเปิดไฟ จนกระทั่งตัวเครื่องสามารถปรับความสว่างภายในให้มีค่าเท่ากับแสงสว่างที่กำหนดไว้ และบันทึกผลการทดลองต่อไปอีกระยะหนึ่งเพื่อดูเสถียรภาพของการรักษาระดับความสว่าง โดยจะทำการทดลองตั้งค่าความสว่างในหลายระดับตั้งแต่ 1,000 ถึง 9,000 ลักซ์ โดยมีความห่างในแต่ละระดับเท่ากับ 1,000 ลักซ์

5.3.2 ผลการทดลองการทำงานของระบบควบคุมแสงสว่าง

จากผลการทดลองดังรูปที่ 5.2 แสดงให้เห็นว่าผลการเปลี่ยนแปลงความเข้มแสงในการทดลองแต่ละครั้งจะใช้เวลาเพียงเล็กน้อยในการปรับค่าแสงสว่างไปสู่ค่าแสงสว่างที่ได้กำหนดไว้ ซึ่งจะใช้เวลาไม่เกิน 20 วินาที สำหรับการรักษาระดับของแสงสว่างภายในตัวเครื่องให้มีเสถียรภาพอยู่ในระดับที่ดี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.2 ผลการทดลองการทำงานของระบบแสงสว่าง

5.4 การทดลองและผลการทดลองระบบควบคุมแสงสว่างในโหมดกลางวันและกลางคืน

5.4.1 การทดลองระบบควบคุมแสงสว่างในโหมดกลางวันและกลางคืน

การทดลองนี้จะเป็นการทดลองควบคุมแสงสว่างในโหมดกลางวันกลางคืน ที่มีการใช้สมการในการหาค่าความสว่างที่ควรจะเป็น ณ ขณะนั้น สำหรับการหาค่าความสว่างจะแบ่งพิจารณาออกเป็น 3 ช่วงเวลา ประกอบด้วย ช่วงแรกจะเป็นช่วงเช้าถึงเที่ยงวัน ช่วงที่สองจะเป็นเวลาหลังเที่ยงวันไปจนถึงช่วงเย็น และช่วงสุดท้ายจะเป็นช่วงเวลากลางคืน ซึ่งช่วงเวลาสุดท้ายจะกำหนดให้แสงมีค่าเป็นศูนย์ กล่าวคือไม่มีการเปิดให้แสงสว่างในช่วงเวลานี้ ส่วนช่วงแรกและช่วงสองนั้นจะถูกแบ่งช่วงเวลาด้วยค่า T_s (เวลาที่เริ่มมีแสงสว่าง) T_p (เวลาที่มีแสงสว่างมากที่สุด) และเวลาที่สิ้นแสงสว่างแล้ว โดยพิจารณากำหนดค่าเวลาทั้งหมดที่เหมาะสมทั้งสามค่า ได้มีการนำข้อมูลทางสถิติที่มีการเก็บบันทึกค่าของแสงในแต่ละวันของกรมอุตุนิยมวิทยามาใช้ในการวิเคราะห์เพื่อกำหนดค่าทั้งสามค่า และได้ค่าเวลาที่เหมาะสมดังนี้ 7.00 น. 12.30 น. และ 18.00 น. ตามลำดับ ดังนั้นในการคำนวณหาอัตราส่วนความสว่างในช่วงเวลาเช้าและบ่ายสามารถหาค่าได้ตามสมการที่ 5.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$I = \begin{cases} L_{set} * \left[\dfrac{\left(\left((T_c - T_s) * 3 + 5\right)\right)}{1000}\right] & , 7.00 < t < 12.30 \\ L_{set} * \left[\dfrac{\left(\left((T_p - T_s) * 3 + 5\right)\right) - \left(\left((T_s - T_p) * 3\right) + 3\right)}{1000}\right] & , 12.30 < t < 18.00 \end{cases} \quad (5.1)$$

เมื่อ

I = ค่าแสงสว่าง (illumination)

L_{set} = ค่าแสงที่ตั้ง (light setting)

t = เวลาปัจจุบัน (current time; Hr:Min)

T_c = เวลาปัจจุบัน (หน่วยวินาที) (current time, second)

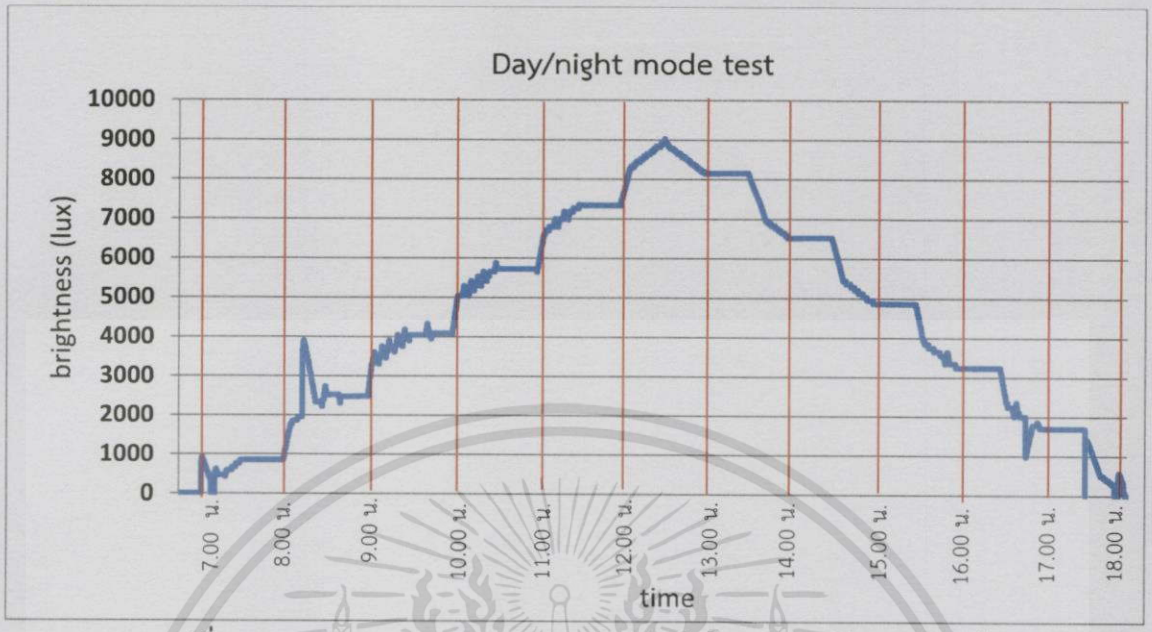
T_s = เวลาเริ่มมีแสงตั้งไว้ที่ 7.00 น (7 * 60 * 60 = 25200 วินาที) (start time, second)

T_p = เวลาแสงสูงสุดตั้งไว้ที่ 12.30 น (((12*60)+30)*60 =45000 วินาที) (peak time.secod)

5.4.2 ผลการทดลองระบบควบคุมแสงสว่างในโหมดกลางวันและกลางคืน

จากการทดลองพบว่า ค่าของแสงสว่างที่มีการเปลี่ยนแปลงตามอัตราส่วนของแสงสว่างที่ได้จากสมการที่ 5.1 นั้น สามารถจำลองค่าความสว่างได้ใกล้เคียงกับแสงตามธรรมชาติค่อนข้างมาก โดยจะเริ่มมีแสงสว่างตั้งแต่ช่วงเวลา 7.00 น. และค่อย ๆ เพิ่มขึ้นตามเวลาไปจนถึงเวลา 12.30 น. ซึ่งเป็นเวลาที่แสงสว่างมีค่ามากที่สุดตามค่าแสงสว่างที่ตั้งไว้ จากนั้นค่าแสงสว่างจะลดลงสิ้นแสงสว่างในเวลา 18.00 น. ดังแสดงในรูปที่ 5.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.3 ผลการทดลองระบบควบคุมแสงสว่างในโหมดกลางวันและกลางคืน

5.5 การทดลองและผลการทดลองการทำงานของระบบควบคุมแสงสว่างร่วมกับอุณหภูมิ

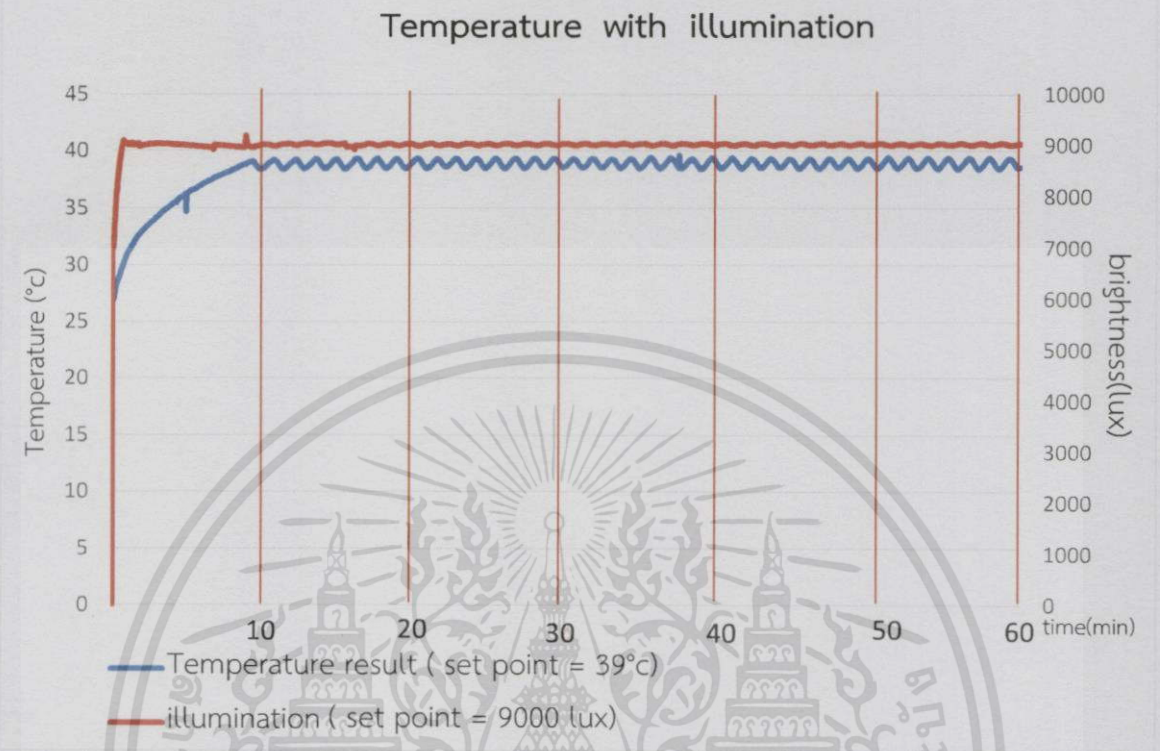
5.5.1 การทดลองหาการทดลองการทำงานของระบบควบคุมแสงสว่างร่วมกับอุณหภูมิ

การทดลองการทำงานของระบบควบคุมแสงสว่างร่วมกันอุณหภูมิ เป็นการทดลองเพื่อทดสอบประสิทธิภาพในการทำงานร่วมกันของระบบแสงสว่างและอุณหภูมิ ว่าในการทำงานร่วมกันตัวเครื่องสามารถทำงานร่วมกันได้หรือไม่ และรักษาค่าแสงสว่างกับอุณหภูมิภายในตัวเครื่องให้มีค่าใกล้เคียงกับค่าที่กำหนด โดยจะทำการทดลองกำหนดค่าความสว่างเท่ากับ 9,000 ลักซ์และใช้อุณหภูมิเริ่มต้นที่ 27 องศาเซลเซียส สำหรับทดสอบทั้งหมด 3 ค่าคือ 39 42 และ 45 องศาเซลเซียส

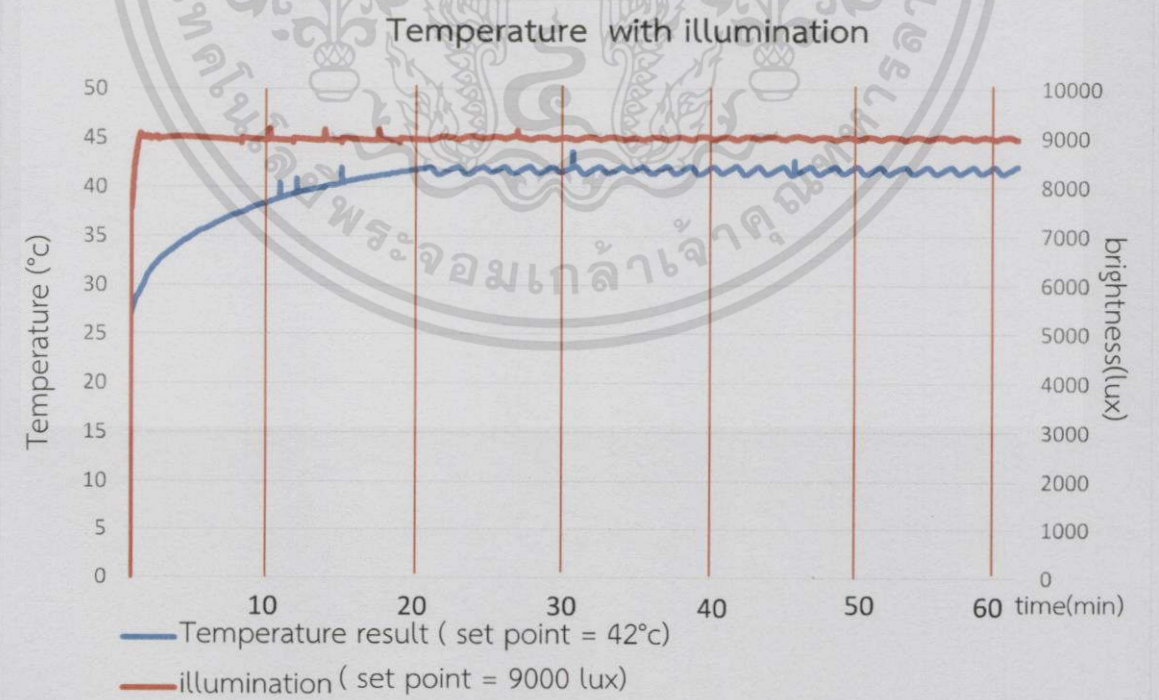
5.5.2 ผลการทดลองการทำงานของระบบควบคุมแสงสว่างร่วมกับอุณหภูมิ

จากผลการทดลองการทำงานของระบบควบคุมแสงสว่างร่วมกับอุณหภูมิ โดยกำหนดให้แสงสว่างมีค่าคงที่เท่ากับ 9,000 ลักซ์ สำหรับอุณหภูมิจะกำหนดไว้ 3 ค่า คือ 39 42 และ 45 องศาเซลเซียส ดังแสดงผลการทดลองตามรูปที่ 5.4 5.5 และ 5.6 ตามลำดับ จากผลการทดลองแสดงให้เห็นว่าตัวเครื่องสามารถทำงานร่วมกันระหว่างแสงสว่างและอุณหภูมิได้ สามารถปรับค่าแสงสว่างและอุณหภูมิภายในตัวเครื่องได้ตามค่าแสงสว่างและอุณหภูมิที่กำหนดไว้ โดยที่อุณหภูมิมากจะต้องใช้เวลาในการปรับค่ามากกว่าที่อุณหภูมิที่มีค่าต่ำกว่า และหลังจากอุณหภูมิภายในตัวเครื่องมีค่าเท่ากับที่ตั้งแล้วสามารถรักษาอุณหภูมิให้มีค่าใกล้เคียงกับค่าที่ตั้งได้ดี ส่วนแสงสว่างในตัวเครื่องหลังจากใช้เวลาในการปรับเพียงเล็กน้อย และสามารถรักษาระดับความสว่างได้ดี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

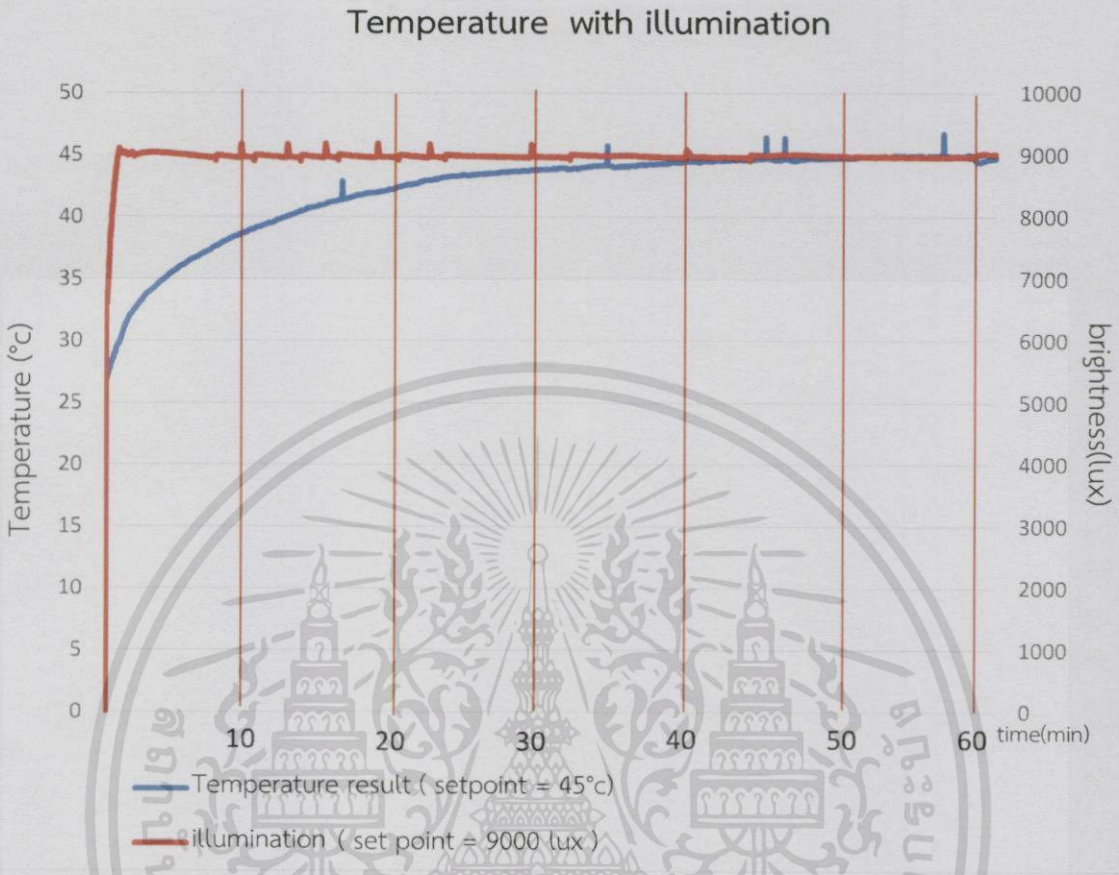


รูปที่ 5.4 ผลการทดลองการทำงานของระบบควบคุมแสงสว่างร่วมกับอุณหภูมิที่ 39 องศาเซลเซียส



รูปที่ 5.5 ผลการทดลองการทำงานของระบบควบคุมแสงสว่างร่วมกับอุณหภูมิที่ 42 องศาเซลเซียส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.6 ผลการทดลองการทำงานของระบบควบคุมแสงสว่างร่วมกับอุณหภูมิที่ 45 องศาเซลเซียส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6
สรุปผล ปัญหาและข้อเสนอแนะ

6.1 สรุปผล

จากการทดลองเครื่องเขย่าแบบควบคุมอุณหภูมิและแสงสว่างนั้น สามารถควบคุมอุณหภูมิได้สูงสุดถึง 45 องศาเซลเซียส ภายใต้อุณหภูมิภายนอกที่มีการควบคุมอุณหภูมิให้คงที่ (ซึ่งในการทดลองอยู่ที่ 27 องศาเซลเซียส) และเงื่อนไขว่าต้องกำหนดค่าความสว่างไว้ที่ค่าสูงสุด (9,000 ลักซ์) และรักษาระดับของอุณหภูมิให้อยู่ช่วงที่ใกล้เคียงกับอุณหภูมิที่ตั้งไว้ได้ สามารถตรวจวัดปริมาณความเข้มของแสงภายในตัวเครื่องได้ รวมถึงสามารถควบคุมและรักษาระดับความเข้มของแสงให้อยู่ในช่วงใกล้เคียงกับความเข้มแสงที่ตั้งไว้โดย โดยสามารถตั้งค่าแสงสว่างได้สูงสุดถึง 9,000 ลักซ์ และสามารถเลือกการทำงานในรูปแบบต่าง ๆ ได้อย่างหลากหลาย เช่น ระบบควบคุมอุณหภูมิ ระบบควบคุมแสงสว่าง ระบบบันทึกค่าอัตโนมัติ และระบบจำลองกลางวันและกลางคืน ซึ่งผลที่ได้เป็นไปตามวัตถุประสงค์ที่กำหนดไว้

6.2 ปัญหาและแนวทางแก้ไข

1. ไม่สามารถรับข้อมูลอุณหภูมิจากตัวเซ็นเซอร์ได้ เนื่องจากสายข้อมูลยาวเกินไป ทำให้ข้อมูลนั้นมีค่าที่ไม่ถูกต้อง จำเป็นต้องมีการเปลี่ยนเซ็นเซอร์จาก แอนะล็อกเซ็นเซอร์ เป็นดิจิตอลเซ็นเซอร์
2. หน้าจอสัมผัสทำการ Reset ตัวเองสาเหตุเกิดจากช่วงเวลารับข้อมูลที่น้อยเกินไป และไม่มีการพักก่อนการทำงานของหน้าจอสัมผัสทำการ Reset ตัวเอง ซึ่งแก้ไขโดยการเพิ่มความล่าช้า (delay) ของการทำงานของหน้าก่อนที่จะทำงาน
3. ซอฟต์แวร์ของอุปกรณ์ที่ใช้งานนั้น ยังมีการปรับปรุงอยู่เป็นระยะ ๆ ทำให้การเขียนโปรแกรมมีการเปลี่ยนแปลงอยู่ตลอดและต้องเปลี่ยนโปรแกรมตามเรื่อย ๆ
4. ข้อมูลที่อาร์ดูโน่ส่งออกมานั้นเป็นแรงดันไฟฟ้าแบบ PWM ซึ่งทำให้แรงดันที่ส่งออกมานั้นมีความไม่เสถียรจึงควรนำ DAC มาใช้งานแทน โดยรับคำสั่งจากอาร์ดูโน่เพื่อส่งไปกำหนดค่าแรงดันไฟฟ้ากระแสตรงแทน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.3 ข้อเสนอแนะ

เครื่องเขย่าสารแบบควบคุมแสงและอุณหภูมินั้นมีราคาที่สูง หากต้องการลดต้นทุนนั้นควรลดค่าใช้จ่ายให้น้อยที่สุด โดยที่หน้าจอสัมผัสที่ได้นำมาทดลองนั้นมีความสะดวกในการออกแบบหน้าจอและมีฟังก์ชันต่าง ๆ ให้ใช้งานค่อนข้างเยอะ แต่มีราคาแพงและฟังก์ชันที่นำมาใช้งานจริง ๆ มีอยู่ไม่มากนัก หากเปลี่ยนมาใช้หน้าจอสัมผัสที่มีราคาถูกและมีประสิทธิภาพมากกว่า จะทำให้ต้นทุนลดลงอีกหากต้องการผลิตในปริมาณมาก และควรปรับปรุงให้เครื่องเขย่ามีการปรับแสงสว่าง และอุณหภูมิได้มากกว่านี้โดยการปรับเปลี่ยนอุปกรณ์ให้มีประสิทธิภาพมากขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

นายเอกชัย มะการ.**เรียนรู้เข้าใจใช้งานไมโครคอนโทรลเลอร์ตระกูล AVR ด้วย Arduino**, 26 มกราคม 2552 บริษัท อีทีที จำกัด พิมพ์ครั้งที่ 1

Arduino. มกราคม 2558.**บอร์ดอาดูโน่เมก้า 2560** .[Online].
Available : http://www.arduino.cc/en/Main/arduinoBoardMega2560

Blogspot. กุมภาพันธ์ 2558. **Data logger เครื่องเก็บข้อมูลอัจฉริยะ**.[Online].
Available : http://limwaltz.blogspot.com/

4dsystems.กุมภาพันธ์ 2558.**ViSi-Genie Writing to Genie Objects Using an Arduino Host** .[Online].
Available : http://www.4dsystems.com.au/appnote/4D-AN-00018/

Eclub Thailand. มีนาคม 2558.**ความรู้เรื่อง I2C** .[Online].
Available : http://www.eclubthai.com/board/index.php?topic=16140.0;wap2

Aimagin . มีนาคม 2558 . **การใช้งานพอร์ตสื่อสาร SPI**.[Online].
Available : http://aimagin.com/blog/spi/?lang=th

Maximintegrated. เมษายน 2558 . **คู่มือใช้งาน DS18B20**.[Online].
Available : http://datasheets.maximintegrated.com/en/ds/DS18B20.pdf

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

Code Program

Code C programming

```
#include <Time.h>
#include <DS1307RTC.h>
#include <SD.h>
#include <SPI.h>
File myFile;
tmElements_t tm;
boolean checkreccord = false;
boolean checkinlogger = false;
boolean checkoutlogger = false;
int tempvalue = 0;
int lightvalue = 0;
int checktempvalue = 0;
int checklightvalue = 0;
short int year_rtc;
short int month_rtc;
short int date_rtc;
short int hour_rtc;
short int min_rtc;
#define time_start 420
#define time_peak 750
#define limit_switch 39
boolean lid_status;
#define relay_1 45//45
#define relay_2 43
#define relay_3 41
#define relay_4 47
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
#define relay_5 49
#define relay_6 51//51
#include <genieArduino.h>
Genie genie;
#define RESETLINE 4
boolean button_data = 0;
boolean button_onoff = 0;
boolean button_on_light = 1;
boolean button_on_temp = 1;
boolean button_daynight = 0;
int button_state = 0;
boolean hold = 0;
boolean toggle = 0;
int addwinb0 = 0;
int real_date = 0;
int addwinb1 = 0;
int real_month = 0;
int addwinb2 = 0;
int real_year = 0;
int addwinb3 = 0;
int real_hour = 0;
int addwinb4 = 0;
int real_min = 0;
unsigned long wait_time = 0;
boolean trigger_onoff = 0;
#include <OneWire.h>
#include <DallasTemperature.h>
#define ONE_WIRE_BUS 2
OneWire oneWire(ONE_WIRE_BUS);
DallasTemperature sensors(&oneWire);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
float temp1;
float temp2;
float tempA;
float tempC;
unsigned int slider_temp;
float set_temp;
unsigned int compare_temp;
int write_temp;
unsigned long wait_temp;
int i = 3;
int buffer_temp;
#define gap_temp 1.0
#include <BH1750FVI.h>
#include <Wire.h>
BH1750FVI LightSensor;
BH1750FVI LightMeter;
unsigned int light_val;
uint16_t lux1;
uint16_t lux2;
unsigned int slider_light;
unsigned int set_light;
unsigned int compare_light;
#define con_light 5
#define gap_light 50
float rate;
boolean trig_light;
int a = 0;

void setup() {
  Serial.begin(14400);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
genie.Begin(Serial);
genie.AttachEventHandler(myGenieEventHandler);
pinMode(RESETLINE, OUTPUT);
digitalWrite(RESETLINE, 1);
delay(100);
digitalWrite(RESETLINE, 0);
delay(4500);
LightSensor.begin();
LightMeter.begin();
LightSensor.SetAddress(Device_Address_H);
LightMeter.SetAddress(Device_Address_L);
LightSensor.SetMode(Continuous_H_resolution_Mode);
LightMeter.SetMode(Continuous_L_resolution_Mode);
pinMode(SS, OUTPUT);
delay(100);
SD.begin(10, 11, 12, 13);
delay(100);
sensors.begin();
slider_temp = 25;
buffer_temp = 25;
set_temp = slider_temp * 1.0;
slider_temp *= 10;
genie.WriteObject(GENIE_OBJ_LED_DIGITS, 0x02, slider_temp);
delay(25);
genie.WriteObject(GENIE_OBJ_LED_DIGITS, 0x06, slider_temp);
delay(25);
genie.WriteObject(GENIE_OBJ_LED_DIGITS, 0x14, slider_temp);
pinMode(limit_switch, INPUT);
pinMode(relay_1, OUTPUT);
pinMode(relay_2, OUTPUT);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  pinMode(relay_3, OUTPUT);
  pinMode(relay_4, OUTPUT);
  pinMode(relay_5, OUTPUT);
  pinMode(relay_6, OUTPUT);
  digitalWrite(relay_1, 1);
  digitalWrite(relay_2, 0);
  digitalWrite(relay_3, 0);
  digitalWrite(relay_4, 0);
  digitalWrite(relay_5, 0);
  digitalWrite(relay_6, 1);
  genie.WriteContrast(9);
  genie.WriteObject(GENIE_OBJ_FORM, 0x00, 0);
}
void loop()
{
  if (millis() <= 2500) {
    wait_time = 0;
  }
  timer();
  start();
  genie.DoEvents();
  if (trigger_onoff == 1) {
    do_button();
    trigger_onoff = 0;
    wait_time = millis() + 1300;
  }
  if (millis() >= wait_time) {
    do_button();
    genie.DoEvents();
    wait_time = millis() + 1200;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  }
  update_display();
  mode_logger();
}
void checklid()
{
  lid_status = digitalRead(limit_switch);
}

void start()
{
  checklid();
  if (lid_status == 0)
  {
    genie.WriteObject(GENIE_OBJ_FORM, 0x01, 0);
    if ( relay_1 == 0 || relay_6 == 0)
    {
      digitalWrite(relay_1, 1);
      digitalWrite(relay_6, 1);
      genie.WriteObject(GENIE_OBJ_WINBUTTON, 0X00, 0);
    }
    button_onoff = 0;
    hold = 0;
    genie.WriteObject(GENIE_OBJ_4DBUTTON, 0x00, button_onoff);
    while (lid_status == 1)
    {
      checklid();
    }
    genie.WriteObject(GENIE_OBJ_FORM, 0x00, 0);
  }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  else
  {
    genie.DoEvents();
  }
}

void get_val()
{
  get_temp();
  get_light();
  genie.DoEvents();
}

void get_temp() {
  sensors.requestTemperatures();
  temp1 = sensors.getTempCByIndex(0);
  temp2 = sensors.getTempCByIndex(1);
  delay(25);
  tempC = (temp1 + temp2) / 2;
  if (tempC > 50 || tempC < 0) {
    genie.DoEvents();
  }
  else {
    tempA = (tempC) * 10.0;
  }
  genie.DoEvents();
}

void get_light() {
  lux1 = LightSensor.GetLightIntensity();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  lux2 = LightMeter.GetLightIntensity();
  light_val = (lux1 + lux2) / 2;
  genie.DoEvents();
}

void com_val()
{
  if (set_light > light_val) {
    compare_light = set_light - light_val;
  }
  else if (set_light <= light_val) {
    compare_light = light_val - set_light;
  }
  delay(25);
  if (set_temp > (tempA/10.0)) {
    compare_temp = set_temp - (tempA/10.0);
  }
  else if (set_temp <= (tempA/10.0)) {
    compare_temp = (tempA/10.0) - set_temp;
  }
}

void update_display()
{
  static unsigned long waitPeriod = 0;
  if (millis() < 500) {
    waitPeriod = 0;
  }
  if (millis() >= waitPeriod)
  {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  if (hold == 0) {
    genie.WriteObject(GENIE_OBJ_LED_DIGITS, 0x03, tempA);
    genie.WriteObject(GENIE_OBJ_LED_DIGITS, 0x01, light_val);
    delay(25);
    genie.WriteObject(GENIE_OBJ_LED_DIGITS, 0x07, tempA);
    genie.WriteObject(GENIE_OBJ_LED_DIGITS, 0x05, light_val);
  }
  else {
    genie.WriteObject(GENIE_OBJ_LED_DIGITS, 0x15, tempA);
    genie.WriteObject(GENIE_OBJ_LED_DIGITS, 0x13, light_val);
  }
  waitPeriod = millis() + 300;
  }
  genie.DoEvents();
}

void control_temp()
{
  get_temp();
  com_val();
  if (compare_temp > gap_temp) {
    if (set_temp < (tempA / 10.0)) {
      digitalWrite(relay_2, 1);
      digitalWrite(relay_3, 1);
    }
    else if (set_temp > (tempA / 10.0)) {
      digitalWrite(relay_2, 0);
      digitalWrite(relay_3, 0);
    }
  }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  else if (compare_temp <= gap_temp) {
    if ( set_temp >= (tempA / 10.0))
    {
      digitalWrite(relay_2, 0);
      digitalWrite(relay_3, 0);
    }
    else
    {
      digitalWrite(relay_2, 0);
      digitalWrite(relay_3, 1);
    }
  }
  genie.DoEvents();
}

void control_light()
{
  get_light();
  com_val();
  if (compare_light >= gap_light) {
    do_light();
  }
  genie.DoEvents();
}

void do_light()
{
  if (set_light < 1000 && set_light != 0 ) {
    if (real_date % 2 == 0) {
      digitalWrite(relay_4, 0);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  digitalWrite(relay_5, 1);
}
else {
  digitalWrite(relay_4, 1);
  digitalWrite(relay_5, 0);
}
if (trig_light == 0) {
  a = 40;
  trig_light = 1;
}
analogWrite(con_light, a);
delay(150);
genie.DoEvents();
get_light();
if (light_val < set_light) {

  analogWrite(con_light, a);
  delay(200);
  genie.DoEvents();
  get_light();
  update_display();
  a += 5;
}
else {
  analogWrite(con_light, a);
  delay(200);
  genie.DoEvents();
  get_light();
  update_display();
  a -= 3;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  }
}
else if (set_light >= 1000 && set_light < 2000) {
  if (real_date % 2 == 0) {
    digitalWrite(relay_4, 1);
    digitalWrite(relay_5, 0);
  }
  else {
    digitalWrite(relay_4, 0);
    digitalWrite(relay_5, 1);
  }
  if (trig_light == 0) {
    a = 65;
    trig_light = 1;
  }
  analogWrite(con_light, a);
  delay(150);
  get_light();
  if (light_val < set_light) {
    analogWrite(con_light, a);
    delay(200);
    genie.DoEvents();
    get_light();
    update_display();
    a += 5;
  }
  else {
    analogWrite(con_light, a);
    delay(200);
    genie.DoEvents();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    get_light();
    update_display();
    a -= 3;
  }
}
else if (set_light >= 2000 && set_light < 3000) {
  digitalWrite(relay_4, 0);
  digitalWrite(relay_5, 0);
  if (trig_light == 0) {
    a = 53;
    trig_light = 1;
  }
  analogWrite(con_light, a);
  delay(150);
  get_light();
  if (light_val < set_light) {
    analogWrite(con_light, a);
    delay(200);
    genie.DoEvents();
    get_light();
    update_display();
    a += 5;
  }
  else {
    analogWrite(con_light, a);
    delay(200);
    genie.DoEvents();
    get_light();
    update_display();
    a -= 3;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  }
}
else if (set_light >= 3000 && set_light < 4000) {
  digitalWrite(relay_4, 0);
  digitalWrite(relay_5, 0);
  if (trig_light == 0) {
    a = 70;
    trig_light = 1;
  }
  analogWrite(con_light, a);
  delay(150);
  get_light();
  if (light_val < set_light) {
    analogWrite(con_light, a);
    delay(200);
    genie.DoEvents();
    get_light();
    update_display();
    a += 5;
  }
  else {
    analogWrite(con_light, a);
    delay(200);
    genie.DoEvents();
    get_light();
    update_display();
    a -= 3;
  }
}
else if (set_light >= 4000 && set_light < 5000) {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    digitalWrite(relay_4, 0);
    digitalWrite(relay_5, 0);
    if (trig_light == 0) {
      a = 90;
      trig_light = 1;
    }
    analogWrite(con_light, a);
    delay(150);
    get_light();
    if (light_val < set_light) {
      analogWrite(con_light, a);
      delay(200);
      genie.DoEvents();
      get_light();
      update_display();
      a += 5;
    }
    else {
      analogWrite(con_light, a);
      delay(200);
      genie.DoEvents();
      get_light();
      update_display();
      a -= 3;
    }
  }
  else if (set_light >= 5000 && set_light < 6000) {
    digitalWrite(relay_4, 0);
    digitalWrite(relay_5, 0);
    if (trig_light == 0) {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    a = 113;
    trig_light = 1;
  }
  analogWrite(con_light, a);
  delay(150);
  get_light();
  if (light_val < set_light) {
    analogWrite(con_light, a);
    delay(200);
    genie.DoEvents();
    get_light();
    update_display();
    a += 5;
  }
  else {
    analogWrite(con_light, a);
    delay(200);
    genie.DoEvents();
    get_light();
    update_display();
    a -= 3;
  }
}
else if (set_light >= 6000 && set_light < 7000) {
  digitalWrite(relay_4, 0);
  digitalWrite(relay_5, 0);
  if (trig_light == 0) {
    a = 135;
    trig_light = 1;
  }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  analogWrite(con_light, a);
  delay(150);
  get_light();
  if (light_val < set_light) {

    analogWrite(con_light, a);
    delay(200);
    genie.DoEvents();
    get_light();
    update_display();
    a += 5;
  }
  else {
    analogWrite(con_light, a);
    delay(200);
    genie.DoEvents();
    get_light();
    update_display();
    a -= 3;
  }
}
else if (set_light >= 7000 && set_light < 8000) {
  digitalWrite(relay_4, 0);
  digitalWrite(relay_5, 0);
  if (trig_light == 0) {
    a = 174;
    trig_light = 1;
  }
  analogWrite(con_light, a);
  delay(150);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    get_light();
    if (light_val < set_light) {
      analogWrite(con_light, a);
      delay(200);
      genie.DoEvents();
      get_light();
      update_display();
      a += 5;
    }
    else {
      analogWrite(con_light, a);
      delay(200);
      genie.DoEvents();
      get_light();
      update_display();
      a -= 3;
    }
  }
  else if (set_light >= 8000 && set_light < 9200) {
    digitalWrite(relay_4, 0);
    digitalWrite(relay_5, 0);
    if (trig_light == 0) {
      a = 220;
      trig_light = 1;
    }
    if(a > 255){
       a = 255;
      }
    analogWrite(con_light, a);
    delay(150);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    get_light();
    if (light_val < set_light) {
      analogWrite(con_light, a);
      delay(200);
      genie.DoEvents();
      get_light();
      update_display();
      if(a == 255){
        genie.DoEvents();
      }
      else{
      a += 5;
      }
    }
    else {
      analogWrite(con_light, a);
      delay(200);
      genie.DoEvents();
      get_light();
      update_display();
      a -= 3;
    }
  }
  else if (set_light == 0) {
    digitalWrite(relay_4, 1);
    digitalWrite(relay_5, 1);
  }
}

void mode_daynight()
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{
  int time_current = (hour_rtc * 60) + min_rtc;
  if (hour_rtc < 7 || hour_rtc >= 18) {
    rate = 0;
  }
  else if (hour_rtc >= 7 && (hour_rtc <= 12 && min_rtc <= 30)) {
    rate = (((time_current - time_start) * 3) + 5) / 1000.00;
  }
  else if ((hour_rtc >= 12 && min_rtc > 30) && hour_rtc < 18) {
    rate = ((((time_peak - time_start) * 3) + 5) - (((time_current - time_peak) * 3) + 3)) /
1000.00;
  }
  set_light = slider_light * rate;
  get_light();
  com_val();
  if (compare_light >= gap_light) {
    do_light();
  }
  genie.DoEvents();
}

void mode_logger()
{
  static unsigned long delaylogger = 0;
  if (button_data == 1 && button_onoff == 1) {
    if (millis() <= 2000) {
      delaylogger = 0;
    }
    genie.DoEvents();
    if (millis() >= delaylogger) {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      com_val();
      logger();
      delaylogger = millis() + 2000;
    }
  }
  else {
    genie.DoEvents();
  }
}

void logger() {
  if (checkinlogger == true) {
    if (checkoutlogger == false) {
      char filename[] = "00002000.CSV";
      RTC.read(tm);
      filename[0] = tm.Day / 10 + '0';
      filename[1] = tm.Day % 10 + '0';
      filename[2] = tm.Month / 10 + '0';
      filename[3] = tm.Month % 10 + '0';
      filename[6] = (tmYearToCalendar(tm.Year) - 2000) / 10 + '0';
      filename[7] = (tmYearToCalendar(tm.Year) - 2000) % 10 + '0';
      myFile = SD.open(filename , FILE_WRITE);
      myFile.println(" ");
      myFile.println(" ");
      myFile.println("Time&Date, Temp, Light, Set Temp, Set Light ");
      myFile.close();
      checkoutlogger = !checkoutlogger ;
    }
    char openfile[] = "00002000.CSV";
    RTC.read(tm);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    openfile[0] = tm.Day / 10 + '0';
    openfile[1] = tm.Day % 10 + '0';
    openfile[2] = tm.Month / 10 + '0';
    openfile[3] = tm.Month % 10 + '0';
    openfile[6] = (tmYearToCalendar(tm.Year) - 2000) / 10 + '0';
    openfile[7] = (tmYearToCalendar(tm.Year) - 2000) % 10 + '0';
    myFile = SD.open(openfile , FILE_WRITE);
    if (myFile) {
      myFile.print(Nowlogger());
      myFile.print(",");
      myFile.println(Allvalue());
      myFile.close();
    }
    }
  }
}
String Nowlogger() {
  String timelogger = "";
  if (RTC.read(tm)) {
    timelogger += tm.Hour;
    timelogger += ":";

    timelogger += tm.Minute;
    timelogger += ":";
    timelogger += tm.Second;
    timelogger += " DAY : ";
    timelogger += tm.Day;
    timelogger += "/";
    timelogger += tm.Month;
    timelogger += "/";
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    timelogger += tmYearToCalendar(tm.Year);
  }
  return timelogger;
}
String Allvalue() {
  String Addvalue = "";
  Addvalue += (tempA / 10);
  Addvalue += ",";
  Addvalue += light_val;
  Addvalue += ",";
  Addvalue += set_temp;
  Addvalue += ",";
  Addvalue += set_light;
  return Addvalue;
}

void cal_button()
{
  if (button_onoff == 0)
  {
    button_state = 0;
  }
  else
  {
    if (button_daynight == 0)
    {
      if ( button_on_temp == 0 && button_on_light == 0)
      {
        button_state = 50;
      }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  else if (button_on_temp == 1 && button_on_light == 0)
  {
    button_state = 51;
  }
  else if (button_on_temp == 0 && button_on_light == 1)
  {
    button_state = 52;
  }
  else
  {
    button_state = 53;
  }
}
else// daynight = 1
{
  if ( button_on_temp == 0 && button_on_light == 0)
  {
    button_state = 54;
  }
  else if (button_on_temp == 1 && button_on_light == 0)
  {
    button_state = 55;
  }
  else if (button_on_temp == 0 && button_on_light == 1)
  {
    button_state = 56;
  }
  else
  {
    button_state = 57;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      }
    }
  }
}

void do_button()
{
  switch (button_state)
  {
    case 0:

      digitalWrite(relay_1, 1);
      digitalWrite(relay_6, 1);
      get_val();
      genie.DoEvents();

      break;

    case 50:

      digitalWrite(relay_1, 0);
      digitalWrite(relay_6, 0);
      digitalWrite(relay_2, 1);
      digitalWrite(relay_3, 1);
      digitalWrite(relay_4, 1);
      digitalWrite(relay_5, 1);
      get_val();
      genie.DoEvents();

      break;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
case 51:

  digitalWrite(relay_1, 0);
  digitalWrite(relay_6, 0);
  digitalWrite(relay_4, 1);
  digitalWrite(relay_5, 1);
  control_temp();
  get_light();
  genie.DoEvents();

  break;

case 52:

  digitalWrite(relay_1, 0);
  digitalWrite(relay_6, 0);
  digitalWrite(relay_2, 1);
  digitalWrite(relay_3, 1);
  control_light();
  get_temp();
  genie.DoEvents();

  break;

case 53:

  digitalWrite(relay_1, 0);
  digitalWrite(relay_6, 0);
  control_temp();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  control_light();
  genie.DoEvents();

  break;

case 54:

  digitalWrite(relay_1, 0);
  digitalWrite(relay_6, 0);
  digitalWrite(relay_2, 1);
  digitalWrite(relay_3, 1);
  get_temp();
  mode_daynight();
  genie.DoEvents();

  break;

case 55:

  digitalWrite(relay_1, 0);
  digitalWrite(relay_6, 0);
  mode_daynight();
  genie.DoEvents();

  break;

case 56:

  digitalWrite(relay_1, 0);
  digitalWrite(relay_6, 0);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      digitalWrite(relay_2, 1);
      digitalWrite(relay_3, 1);
      get_temp();
      mode_daynight();
      genie.DoEvents();

      break;

    case 57:

      digitalWrite(relay_1, 0);
      digitalWrite(relay_6, 0);
      control_temp();
      mode_daynight();
      genie.DoEvents();

      break;
  }
}

void timer()
{
  static unsigned long wait_settime = millis();
  static unsigned long suggest_time = millis();
  genie.DoEvents();
  if (millis() < 10000) {
    wait_settime = 0;
    suggest_time = 0;
  }
  if (millis() >= wait_settime)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{
  if (RTC.read(tm))
  {
    date_rtc = tm.Day;
    real_date = date_rtc;
    genie.WriteObject(GENIE_OBJ_LED_DIGITS, 0x08, date_rtc);
    genie.WriteObject(GENIE_OBJ_LED_DIGITS, 0x16, date_rtc);
    month_rtc = tm.Month;
    real_month = month_rtc;
    genie.WriteObject(GENIE_OBJ_LED_DIGITS, 0x09, month_rtc);
    delay(25);
    genie.WriteObject(GENIE_OBJ_LED_DIGITS, 0x17, month_rtc);
    year_rtc = tmYearToCalendar(tm.Year) - 2000;
    real_year = year_rtc;
    genie.WriteObject(GENIE_OBJ_LED_DIGITS, 0x0A, year_rtc);
    delay(25);
    genie.WriteObject(GENIE_OBJ_LED_DIGITS, 0x18, year_rtc);
    hour_rtc = tm.Hour;
    real_hour = hour_rtc;
    genie.WriteObject(GENIE_OBJ_LED_DIGITS, 0x0B, hour_rtc);
    delay(25);
    genie.WriteObject(GENIE_OBJ_LED_DIGITS, 0x19, hour_rtc);
    min_rtc = tm.Minute;
    real_min = min_rtc;
    genie.WriteObject(GENIE_OBJ_LED_DIGITS, 0x0C, min_rtc);
    delay(25);
    genie.WriteObject(GENIE_OBJ_LED_DIGITS, 0x1A, min_rtc);
    if (millis() >= suggest_time)
    {
      addwinb0 = date_rtc;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      real_date = date_rtc;
      genie.WriteObject(GENIE_OBJ_LED_DIGITS, 0x0D, addwinb0);
      addwinb1 = month_rtc;
      real_month = month_rtc;
      genie.WriteObject(GENIE_OBJ_LED_DIGITS, 0x0E, addwinb1);
      addwinb2 = year_rtc;
      real_year = year_rtc;
      genie.WriteObject(GENIE_OBJ_LED_DIGITS, 0x0F, addwinb2);
      addwinb3 = hour_rtc;
      real_hour = hour_rtc;
      genie.WriteObject(GENIE_OBJ_LED_DIGITS, 0x10, addwinb3);
      addwinb4 = min_rtc;
      real_min = min_rtc;
      genie.WriteObject(GENIE_OBJ_LED_DIGITS, 0x11, addwinb4);
      suggest_time += 600000;
    }
    wait_settime += 60000;
  }
 }
genie.DoEvents();
}
void myGenieEventHandler(void)
{
  genieFrame Event;
  genie.DequeueEvent(&Event);
  if (Event.reportObject.cmd == GENIE_REPORT_EVENT)
  {
    if (Event.reportObject.object == GENIE_OBJ_4DBUTTON)
    {
      if (Event.reportObject.index == 0x00)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  {
    button_onoff = 1;
    hold = 1;
    trigger_onoff = 1;
    genie.WriteObject(GENIE_OBJ_FORM, 0x06, 0);
  }
  else if (Event.reportObject.index == 0x01)
  {
    button_on_light  = genie.GetEventData(&Event);
  }
  else if (Event.reportObject.index == 0x02)
  {
    button_on_temp  = genie.GetEventData(&Event);
  }
  else if (Event.reportObject.index == 0x03)
  {
    button_onoff = 0;
    hold = 0;
    trigger_onoff = 1;
    trig_light = 0;
    genie.WriteObject(GENIE_OBJ_LED_DIGITS, 0x15, 0);
    genie.WriteObject(GENIE_OBJ_LED_DIGITS, 0x13, 0);
    genie.WriteObject(GENIE_OBJ_FORM, 0x02, 0);
  }
}
else if (Event.reportObject.object == GENIE_OBJ_WINBUTTON)
{
  if (Event.reportObject.index == 0x01)
  {
    button_data = genie.GetEventData(&Event);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  checkreccord = !checkreccord;
  if (checkreccord == true ) {
    genie.WriteObject(GENIE_OBJ_LED, 0x03, button_data);
    genie.WriteObject(GENIE_OBJ_LED, 0x04, button_data);
    checkinlogger = true;
  }
  else if (checkreccord == false ) {
    genie.WriteObject(GENIE_OBJ_LED, 0x03, button_data);
    genie.WriteObject(GENIE_OBJ_LED, 0x04, button_data);
    checkinlogger = false;
    checkoutlogger = false;
  }
}
else if (Event.reportObject.index == 0x03)
{
  button_daynight = genie.GetEventData(&Event);
  genie.WriteObject(GENIE_OBJ_LED, 0x02, button_daynight);
  genie.WriteObject(GENIE_OBJ_LED, 0x05, button_daynight);
}
else if (Event.reportObject.index == 0x07)
{
  slider_light += 100;
  genie.WriteObject(GENIE_OBJ_SLIDER, 0x00, (slider_light / 100));
  genie.WriteObject(GENIE_OBJ_LED_DIGITS, 0x04, slider_light);
  delay(25);
  genie.WriteObject(GENIE_OBJ_LED_DIGITS, 0x00, slider_light);
  delay(25);
  genie.WriteObject(GENIE_OBJ_LED_DIGITS, 0x12, slider_light);
  set_light = slider_light;
  trig_light = 0;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
}
else if (Event.reportObject.index == 0x08)
{
  slider_light -= 100;
  genie.WriteObject(GENIE_OBJ_SLIDER, 0x00, (slider_light / 100));
  genie.WriteObject(GENIE_OBJ_LED_DIGITS, 0x04, slider_light);
  delay(25);
  genie.WriteObject(GENIE_OBJ_LED_DIGITS, 0x00, slider_light);
  delay(25);
  genie.WriteObject(GENIE_OBJ_LED_DIGITS, 0x12, slider_light);
  set_light = slider_light;
  trig_light = 0;
}
else if (Event.reportObject.index == 0x0A)
{
  i++;
  slider_temp += 5;
  set_temp += 0.5;
  if (i == 5) {
    i = 3;
    int move_temp = buffer_temp + 1;
    genie.WriteObject(GENIE_OBJ_SLIDER, 0x01, move_temp);
    buffer_temp = move_temp;
  }
  genie.WriteObject(GENIE_OBJ_LED_DIGITS, 0x06, slider_temp);
  delay(25);
  genie.WriteObject(GENIE_OBJ_LED_DIGITS, 0x02, slider_temp);
  delay(25);
  genie.WriteObject(GENIE_OBJ_LED_DIGITS, 0x14, slider_temp);
  set_temp = (slider_temp / 10.0);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
}
else if (Event.reportObject.index == 0x0B)
{
  i--;
  slider_temp -= 5;
  set_temp -= 0.5;
  if (i == 1) {
    i = 3;
    int move_temp = buffer_temp - 1;
    genie.WriteObject(GENIE_OBJ_SLIDER, 0x01, move_temp);
    buffer_temp = move_temp;
  }
  genie.WriteObject(GENIE_OBJ_LED_DIGITS, 0x06, slider_temp);
  delay(25);
  genie.WriteObject(GENIE_OBJ_LED_DIGITS, 0x02, slider_temp);
  delay(25);
  genie.WriteObject(GENIE_OBJ_LED_DIGITS, 0x14, slider_temp);
  set_temp = (slider_temp / 10.0);
}
else if (Event.reportObject.index == 0x0C)
{
  if (real_month == 4 || real_month == 6 || real_month == 9 || real_month == 11)
  {
    addwinb0++;
    if (addwinb0 > 30)
    {
      addwinb0 = 0;
      real_date = 0;
    }
    real_date = addwinb0;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      genie.WriteObject(GENIE_OBJ_LED_DIGITS, 0x0D, real_date);
    }
    else if (real_month == 1 || real_month == 3 || real_month == 5 || real_month == 7 ||
real_month == 8 || real_month == 10 || real_month == 12)
    {
      addwinb0++;
      if (addwinb0 > 31)
      {
        addwinb0 = 1;
        real_date = 1;
      }
      real_date = addwinb0;
      genie.WriteObject(GENIE_OBJ_LED_DIGITS, 0x0D, real_date);
    }
    else
    {
      addwinb0++;
      if (addwinb0 > 28)
      {
        addwinb0 = 1;
        real_date = 1;
      }
      real_date = addwinb0;
      genie.WriteObject(GENIE_OBJ_LED_DIGITS, 0x0D, real_date);
    }
  }
  else if (Event.reportObject.index == 0x0D)
  {
    tm.Hour = real_hour;
    tm.Minute = real_min;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  tm.Second = 0;
  tm.Day = real_date;
  tm.Month = real_month;
  tm.Year = CalendarYrToTm(real_year + 2000);
  RTC.write(tm);
  genie.WriteObject(GENIE_OBJ_LED_DIGITS, 0x08, real_date);
  delay(10);
  genie.WriteObject(GENIE_OBJ_LED_DIGITS, 0x09, real_month);
  delay(10);
  genie.WriteObject(GENIE_OBJ_LED_DIGITS, 0x0A, real_year);
  delay(10);
  genie.WriteObject(GENIE_OBJ_LED_DIGITS, 0x0B, real_hour);
  delay(10);
  genie.WriteObject(GENIE_OBJ_LED_DIGITS, 0x0C, real_min);
}
else if (Event.reportObject.index == 0x0F)
{
  addwinb1++;
  if (addwinb1 > 12)
  {
    addwinb1 = 1;
    real_month = 1;
  }
  real_month = addwinb1;
  genie.WriteObject(GENIE_OBJ_LED_DIGITS, 0x0E, real_month);
}
else if (Event.reportObject.index == 0x10)
{
  addwinb2++;
  if (addwinb2 > 25)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    {
      real_year = 15;
      addwinb2 = 15;
    }
    real_year = addwinb2;
    genie.WriteObject(GENIE_OBJ_LED_DIGITS, 0x0F, real_year);
  }
  else if (Event.reportObject.index == 0x11)
  {
    addwinb3++;
    if (addwinb3 > 23)
    {
      real_hour = 0;
      addwinb3 = 0;
    }
    real_hour = addwinb3;
    genie.WriteObject(GENIE_OBJ_LED_DIGITS, 0x10, real_hour);
  }
  else if (Event.reportObject.index == 0x12)
  {
    addwinb4++;
    if (addwinb4 > 59)
    {
      real_min = 0;
      addwinb4 = 0;
    }
    real_min = addwinb4;
    genie.WriteObject(GENIE_OBJ_LED_DIGITS, 0x11, real_min);
  }
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  else if (Event.reportObject.object == GENIE_OBJ_SLIDER)
  {
    if (Event.reportObject.index == 0x00)
    {
      slider_light = genie.GetEventData(&Event);
      slider_light *= 100;
      genie.WriteObject(GENIE_OBJ_LED_DIGITS, 0x04, slider_light);
      delay(25);
      genie.WriteObject(GENIE_OBJ_LED_DIGITS, 0x00, slider_light);
      delay(25);
      genie.WriteObject(GENIE_OBJ_LED_DIGITS, 0x12, slider_light);
      set_light = slider_light;
      trig_light = 0;
    }
    else if (Event.reportObject.index == 0x01){
      slider_temp = genie.GetEventData(&Event);
      buffer_temp = slider_temp*1;
      slider_temp += 25;
      set_temp = slider_temp * 1.0;
      slider_temp *= 10;
      genie.WriteObject(GENIE_OBJ_LED_DIGITS, 0x06, slider_temp);
      delay(25);
      genie.WriteObject(GENIE_OBJ_LED_DIGITS, 0x02, slider_temp);
      delay(25);
      genie.WriteObject(GENIE_OBJ_LED_DIGITS, 0x14, slider_temp);
    }
   }
  }
  cal_button();
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้