



ระบบควบคุมอุณหภูมิภายในโรงเรือนอัจฉริยะ
Temperature Control System in Intelligent House

จัดทำโดย

นายกฤษฎา	ขาวมู่
นายณรงค์	มูลศิริ

โครงการนี้เป็นส่วนหนึ่งของวิชาโครงการ หลักสูตรประกาศนียบัตรวิชาชีพชั้นสูง
สาขาวิชาเทคโนโลยีสารสนเทศ
วิทยาลัยเทคโนโลยีอรรถวิทย์พัฒนการ
ปีการศึกษา 2562

ระบบควบคุมอุณหภูมิภายในโรงเรือนอัจฉริยะ
Temperature Control System in Intelligent House

จัดทำโดย

นายกฤษฎา	ขาวมณี
นายณรงค์	มุลศิริ

โครงการนี้เป็นส่วนหนึ่งของวิชาโครงการ หลักสูตรประกาศนียบัตรวิชาชีพชั้นสูง
สาขาวิชาเทคโนโลยีสารสนเทศ
วิทยาลัยเทคโนโลยีอรรณพวิทยพัฒน์
ปีการศึกษา 2562

COPYRIGHT 2019

COLLEGE OF INFORMATION TECHNOLOGY

ATTAWIT COMMERCIAL TECHNOLOGY COLLEGE



ชื่อโครงการภาษาไทย

ระบบควบคุมอุณหภูมิภายในโรงเรือนอัจฉริยะ

ชื่อโครงการภาษาอังกฤษ

Temperature Control System in Intelligent House

โดย 1. นายกฤษฎา ขาวนู่ รหัสประจำตัว 41269
2. นายณรงค์ มูลศิริ รหัสประจำตัว 40978

.....
คณะกรรมการอนุมัติให้เอกสารโครงการฉบับนี้เป็นส่วนหนึ่งของการศึกษา วิชาโครงการตามหลักสูตรประกาศนียบัตรวิชาชีพชั้นสูง (ปวส.) สาขาวิชาเทคโนโลยีสารสนเทศ วิทยาลัยเทคโนโลยีอรรถวิทย์พัฒนวิชาการ (ATC)

.....
(อาจารย์คุณานนท์ สุขเกษม)

อาจารย์ที่ปรึกษา

.....
(อาจารย์คุณานนท์ สุขเกษม)

หัวหน้าสาขาวิชาเทคโนโลยีสารสนเทศ

บทคัดย่อ

หัวข้อโครงการ ระบบควบคุมอุณหภูมิภายในโรงเรือนอัจฉริยะ
Temperature Control System in Intelligent House

ผู้จัดทำโครงการ 1. นายกฤษฎา ขาวบู่ รหัสประจำตัว 41269
2. นายณรงค์ มุลศิริ รหัสประจำตัว 40978

ชื่ออาจารย์ที่ปรึกษา อาจารย์คุณานนท์ สุขเกษม
สาขาวิชา เทคโนโลยีสารสนเทศ
สถาบัน วิทยาลัยเทคโนโลยีอรรถวิทย์พัฒนชยการ ปีการศึกษา 2562

บทคัดย่อ

โครงการเล่มนี้จัดทำขึ้นเพื่อนำความรู้ที่ได้ศึกษามาในสาขาวิชาเทคโนโลยีสารสนเทศมาประยุกต์ใช้ในการจัดทำโครงการเรื่อง “ระบบควบคุมอุณหภูมิภายในโรงเรือนอัจฉริยะ” โดยจะเน้นการใช้โปรแกรม Arduino ในการจัดทำเป็นหลัก

การกระทำให้ระบบควบคุมอุณหภูมิภายในโรงเรือนอัจฉริยะนั้นได้จัดทำโดยโปรแกรม Arduino ซึ่งเป็นโปรแกรมที่มีไว้สำหรับการเขียนโปรแกรมเพื่อป้อนข้อมูลลงแผงวงจร อีกทั้งยังมีการนำมอเตอร์เซอร์โว มาช่วยในเรื่องการระบายความร้อนโดยการเปิดและปิดหลังคาโรงเรือน ซึ่งหากค่าแสงเกินหรือต่ำกว่าที่กำหนด ให้หลังคาทำการเปิดปิดเองอัตโนมัติ , ระบบระบายความร้อนด้วยพัดลม โดยใช้เซนเซอร์วัดอุณหภูมิมาวัดอุณหภูมิ หากอุณหภูมิสูงหรือต่ำกว่าที่กำหนดไว้ ให้พัดลมเปิด-ปิดทำงานเองอัตโนมัติ แล้วนำข้อมูลมาแสดงผลผ่านจอ LCD ช่วยให้เราสามารถรับรู้ค่าอุณหภูมิและค่าแสงของสภาพแวดล้อมต่าง ๆ ภายในระบบควบคุมโรงเรือนเพาะปลูกเพื่อเป็นการลดสภาพความแปรปรวนของดินฟ้าอากาศและเพื่อเพิ่มผลผลิตที่มากขึ้น

จากการศึกษาและสืบค้นข้อมูลก่อนการลงมือปฏิบัติทางคณะผู้จัดทำได้รับความรู้ในเรื่องต่าง ๆ และยังได้รับประสบการณ์นอกห้องเรียนเป็นอย่างมากทางคณะผู้จัดทำจึงหวังเป็นอย่างมากว่าโครงการเล่มนี้จะมีประโยชน์ต่อผู้พบเห็นไม่มากก็น้อยตามความเหมาะสม

กิตติกรรมประกาศ

โครงการฉบับนี้สำเร็จลุล่วงไปได้เป็นอย่างดีก็เพราะได้รับความเมตตาจากอาจารย์คุณานนท์ สุขเกษม ที่ได้ให้คำแนะนำและคำปรึกษาเกี่ยวกับการสร้างระบบควบคุมอุณหภูมิภายในโรงเรือนอัจฉริยะ มาโดยตลอด ผู้ทำโครงการรู้สึกซาบซึ้งในความอนุเคราะห์จากท่านเป็นอย่างมาก และกราบขอบพระคุณเป็นอย่างสูง

ผู้ทำโครงการขอกราบขอบพระคุณ คุณพ่อ คุณแม่ และเพื่อนๆ พี่ๆ ที่ให้กำลังใจและให้โอกาสบุตรได้รับการศึกษาในระดับต่าง ๆ จนกระทั่งได้สำเร็จการศึกษาระดับประกาศนียบัตรวิชาชีพชั้นสูง รวมทั้งคณะอาจารย์ทุกท่านในสาขาวิชาเทคโนโลยีสารสนเทศที่ได้ประสิทธิ์ประสาทวิชาความรู้และ คำสั่งสอนให้กับผู้ทำโครงการในการเรียนทุกระดับชั้น

ขอขอบพระคุณครอบครัวที่ให้การช่วยเหลือในทุก ๆ ด้าน จนทำให้ประสบความสำเร็จลุล่วงไปได้ ด้วยดี ขอบขอบคุณเพื่อนๆ พี่ๆ ทุกคน ในสาขาวิชาเทคโนโลยีสารสนเทศ วิทยาลัยเทคโนโลยี อรรถวิทย์พัฒนชการ ที่คอยให้การช่วยเหลือการทำโครงการฉบับนี้

สุดท้ายความรู้และประโยชน์ที่ได้รับจากโครงการฉบับนี้ ผู้ทำโครงการ ขอมอบความดีความชอบ ที่ได้นี้ให้แก่ผู้มีพระคุณทุกท่าน

นายกฤษฎา ขาวมู

นายณรงค์ มูลศิริ

สารบัญ

หน้า

บทคัดย่อภาษาไทย	I
กิตติกรรมประกาศ.....	II
สารบัญ	III
สารบัญตาราง	V
สารบัญรูป	VI
บทที่ 1 บทนำ.....	1
1.1 ความเป็นมาและความสำคัญของปัญหา.....	1
1.2 วัตถุประสงค์ของโครงการ	2
1.3 ขอบเขตของโครงการ	2
1.4 ขั้นตอนการดำเนินงาน	2
1.5 ระยะเวลาการทำโครงการ	5
1.6 ประโยชน์ที่คาดว่าจะได้รับ	6
1.7 งบประมาณที่ใช้ในการทำโครงการ	6
บทที่ 2 เอกสารและทฤษฎีที่เกี่ยวข้อง	7
2.1 ไมโครคอนโทรลเลอร์ Node MCU	7
2.2 บอร์ด Arduino.....	9
2.3 LCD Display	11
2.4 Relay 4 Chanel.....	17
2.5 BH1750 Lux Sensor.....	20
2.6 Servo Motor	21
2.7 Temperature Sensor	25
2.8 หลักการเขียนโปรแกรม Arduino	27
บทที่ 3 วิธีการดำเนินโครงการ	28
3.1 การวางแผนและการเตรียมการ.....	28
3.2 การออกแบบชิ้นงาน.....	31
3.3 การดำเนินการสร้างระบบควบคุมอุณหภูมิภายในโรงเรือนอัจฉริยะ	32

สารบัญ (ต่อ)

หน้า

บทที่ 4 ผลการศึกษา	35
4.1 การทดสอบวัดอุณหภูมิและระบบระบายความร้อนเองอัตโนมัติด้วยพัดลม	35
4.2 การทดสอบวัดค่าแสงและระบบระบายความร้อนเองอัตโนมัติ ด้วยมอเตอร์เซอร์โว.....	37
บทที่ 5 สรุป อภิปรายผล และข้อเสนอแนะ	41
5.1 วัตถุประสงค์ของโครงการ	41
5.2 ประโยชน์ที่ได้รับจากการศึกษา	41
5.3 ปัญหาที่พบในการดำเนินโครงการ	41
5.4 ผลการดำเนินโครงการ	42
5.5 อภิปรายผล	42
5.6 ข้อเสนอแนะ.....	42
บรรณานุกรม	43
ภาคผนวก ก.....	44
แบบเสนอร่างโครงการ	44
ภาคผนวก ข	45
รายงานผลความก้าวหน้าโครงการ	45
ภาคผนวก ค	46
ประวัติผู้เขียน	46

สารบัญตาราง

ตารางที่	หน้า
1.1 ตารางแผนการดำเนินงาน	5
2.1 แสดงค่าโมดูลเบอร์ PCF8574	12
2.2 แสดงโมดูลเบอร์ PCF8574A	12
2.3 แสดงขาที่ใช้ในการเชื่อมต่อของ Relay Module 4 Channels.....	20
2.4 แสดงข้อผิดพลาดของบอร์ด	21
3.1 ตารางแผนการดำเนินงาน	30
4.1 ผลการทดลองการวัดอุณหภูมิขณะอากาศปกติ.....	36
4.2 ผลการทดลองการวัดอุณหภูมิขณะอากาศร้อน	37
4.3 ผลการทดลองการวัดค่าแสงช่วงเวลา 8.00-12.00 น.	39
4.4 ผลการทดลองการวัดค่าแสงช่วงเวลา 12.00-15.00 น.	39

สารบัญรูป

รูปที่	หน้า
1.1 แผนภาพ Flowchart ขั้นตอนการดำเนินงาน	3
1.2 ภาพบอร์ดNodeMCU v.2	8
2.2 แสดงตำแหน่งพินและขาของ NodeMCU v.2	8
2.3 Arduino RS232 ต้นแบบของอาคุยโน้	10
2.4 Layout & Pin out Arduino Board (Model: Arduino UNO R3)	10
2.5 แสดงตัวอย่าง LCD.....	11
2.6 แสดงการต่อ LCD แบบขนาน	12
2.7 แสดงการต่อ LCD แบบอนุกรม	13
2.8 แสดงการเพิ่ม Library ในโปรแกรม Arduino IDE	13
2.9 แสดงไลบรารีได้ถูกเพิ่มเข้ามาแล้ว	14
2.10 การเปลี่ยนหมายเลขประจำตัว (I2C Address)	15
2.11 แสดงตัวอย่าง Relay	17
2.12 สัญลักษณ์ในวงจรไฟฟ้าของรีเลย์	17
2.13 แสดงตัวอย่าง SPST และ SPDT	18
2.14 แสดงตัวอย่าง DPST และ DPDT.....	18
2.15 แสดงตัวอย่าง Relay Module 4 Channels	18
2.16 GY-302 BH1750FVI เซนเซอร์วัดค่าแสง	21
2.17 เซอร์โวมอเตอร์ (Servo Motor)	21
2.18 ส่วนประกอบของเซอร์โวมอเตอร์.....	22
2.19 ส่วนประกอบภายใน RC Servo Motor.....	23
2.20 บล็อกไดอะแกรมของ Servo Motor	23
2.21 Remote Control RC Servo Motor	24
2.22 มุมหรือองศาหมุนขึ้นอยู่กับความกว้างของสัญญาณพัลส์	25
2.23 RC Servo Motor หมุนไปที่ 45 องศา.....	25
2.24 DHT-11 วัดอุณหภูมิและความชื้น	26
3.1 ออกแบบภายนอกโรงเรือน	31
3.2 โครงสร้างฐานโรงเรือน.....	31
3.3 โครงสร้างฐานโรงเรือนขึ้นส่วนด้านหน้าและหลัง.....	31
3.4 โครงสร้างโรงเรือนขึ้นส่วนด้านข้าง.....	32

สารบัญรูป(ต่อ)

รูปที่	หน้า
3.5 โครงสร้างโรงเรือนชั้นส่วนหลังคาโรงเรือน	32
3.6 ขั้นตอนการสร้างโรงเรือน	33
3.7 ขั้นตอนการประกอบและติดตั้ง	33
3.8 ขั้นตอนการต่อวงจร	34
3.9 ขั้นตอนการเขียนโปรแกรมคำสั่ง	34
4.1 อ่านค่าที่ความชื้นในดินจาก Serial Monitor	35
4.2 อ่านค่าอุณหภูมิในอากาศจาก LCD.....	36
4.3 ภาพแสดงการทำงานของพัลลภระบายความร้อน	36
4.4 อ่านค่าแสงจาก Serial Monitor	38
4.5 อ่านค่าแสงจาก LCD	38
4.6 ภาพแสดงการทำงานของหลังคาโรงเรือน	38

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ปัจจุบันสภาพอากาศค่อนข้างมีความแปรปรวนและยากที่จะควบคุม ส่งผลกระทบทั้งต่อการดำเนินชีวิตของมนุษย์ สัตว์ สิ่งแวดล้อม รวมถึงพืช แต่ข้อจำกัดของพืชคือไม่สามารถเลือกสภาพแวดล้อมที่เหมาะสมในการเจริญเติบโตได้อย่างมนุษย์หรือสัตว์ ดังนั้นเพื่อควบคุมปัจจัยด้านสภาพแวดล้อมในการเพาะปลูกพืชให้มีความเหมาะสมนั้นจึงมีการพัฒนาการปลูกพืชสู่การปลูกในระบบโรงเรือน แต่ปัญหาการปลูกพืชในโรงเรือนในประเทศไทยคือ “อุณหภูมิหรือความร้อนสะสมภายในโรงเรือน” โดยเฉพาะโรงเรือนที่ไม่มี การระบายความร้อน มีโอกาสที่อุณหภูมิภายในจะสูงถึง 50 องศาเซลเซียสซึ่งมีผลต่อการเจริญเติบโตของพืชอย่างมาก ดังนั้นการเลือกโรงเรือนจึงมีความสำคัญเนื่องจากโรงเรือนแต่ละแบบมีความสามารถในการระบายอากาศและการลดอุณหภูมิได้แตกต่างกัน

รศ.ดร.ธรรมศักดิ์ ทองเกตุ อาจารย์ภาควิชาพืชสวน คณะเกษตร มหาวิทยาลัยเกษตรศาสตร์ กล่าวว่า ประเทศไทยตั้งอยู่ในเขตร้อน การปลูกพืชจึงมีโอกาที่จะเกิดโรคและแมลงศัตรูรบกวนได้ตลอด โดยเฉพาะการปลูกพืชในโรงเรือนที่ไม่สามารถควบคุมสภาพแวดล้อมต่าง ๆ ได้ การผลิตพืชในระบบปิด หรือการปลูกพืชในโรงเรือน จึงเป็นคำตอบที่จะช่วยให้เกษตรกรสามารถผลิตพืชได้ทั้งปริมาณ คุณภาพ และปลอดภัยได้อย่างต่อเนื่องตลอดทั้งปี ภายใต้การควบคุมปัจจัยแวดล้อมที่ไม่เหมาะสม เช่น ฝน พายุ ความแปรปรวนของสภาพอากาศ ป้องกันการระบาดของโรคและแมลงศัตรูลดการใช้สารเคมี นอกจากนี้การปลูกพืชในโรงเรือนยังช่วยเพิ่มประสิทธิภาพในการให้น้ำ (Fertigation System) เป็นระบบการผลิตพืชที่เป็นมิตรกับสิ่งแวดล้อม

ผู้วิจัยมีแนวคิดที่จะนำเอาเทคโนโลยีไอโอที (IOT) มาประดิษฐ์ระบบควบคุมอุณหภูมิภายในโรงเรือนอัจฉริยะ โดยนำระบบของมอเตอร์เซอร์โว มาช่วยในเรื่องการระบายความร้อนโดยการเปิดและปิดหลังคาโรงเรือน โดยหากค่าแสงเกินกว่าที่กำหนดให้หลังคาทำการเปิดเองอัตโนมัติ และหากค่าแสงต่ำกว่าที่กำหนดให้หลังคาทำการปิดเองอัตโนมัติ , ระบบระบายความร้อนด้วยพัดลม โดยใช้เซนเซอร์วัดอุณหภูมิมาวัดอุณหภูมิ หากอุณหภูมิสูงกว่าที่กำหนดไว้ ให้พัดลมทำงานเองอัตโนมัติ และหากอุณหภูมิต่ำกว่าที่กำหนดไว้ ให้พัดลมปิดการทำงานเองอัตโนมัติ แล้วนำข้อมูลมาแสดงผลผ่านจอ LCD ช่วยให้ผู้สามารถรับรู้ค่าอุณหภูมิและค่าแสงของสภาพแวดล้อมต่าง ๆ ภายในระบบควบคุมโรงเรือนเพาะปลูกเพื่อเป็นการลดสภาพความแปรปรวนของดินฟ้าอากาศและเพื่อเพิ่มผลผลิตที่มากขึ้น โดยที่ความร้อนภายในระบบควบคุมโรงเรือนเพาะปลูกจะอยู่ในช่วงที่เหมาะสมเพื่อให้การดูแลระบบสมาร์ตฟาร์มจำลองเป็นไปอย่างอัตโนมัติและมีประสิทธิภาพ

1.2 วัตถุประสงค์ของโครงการ

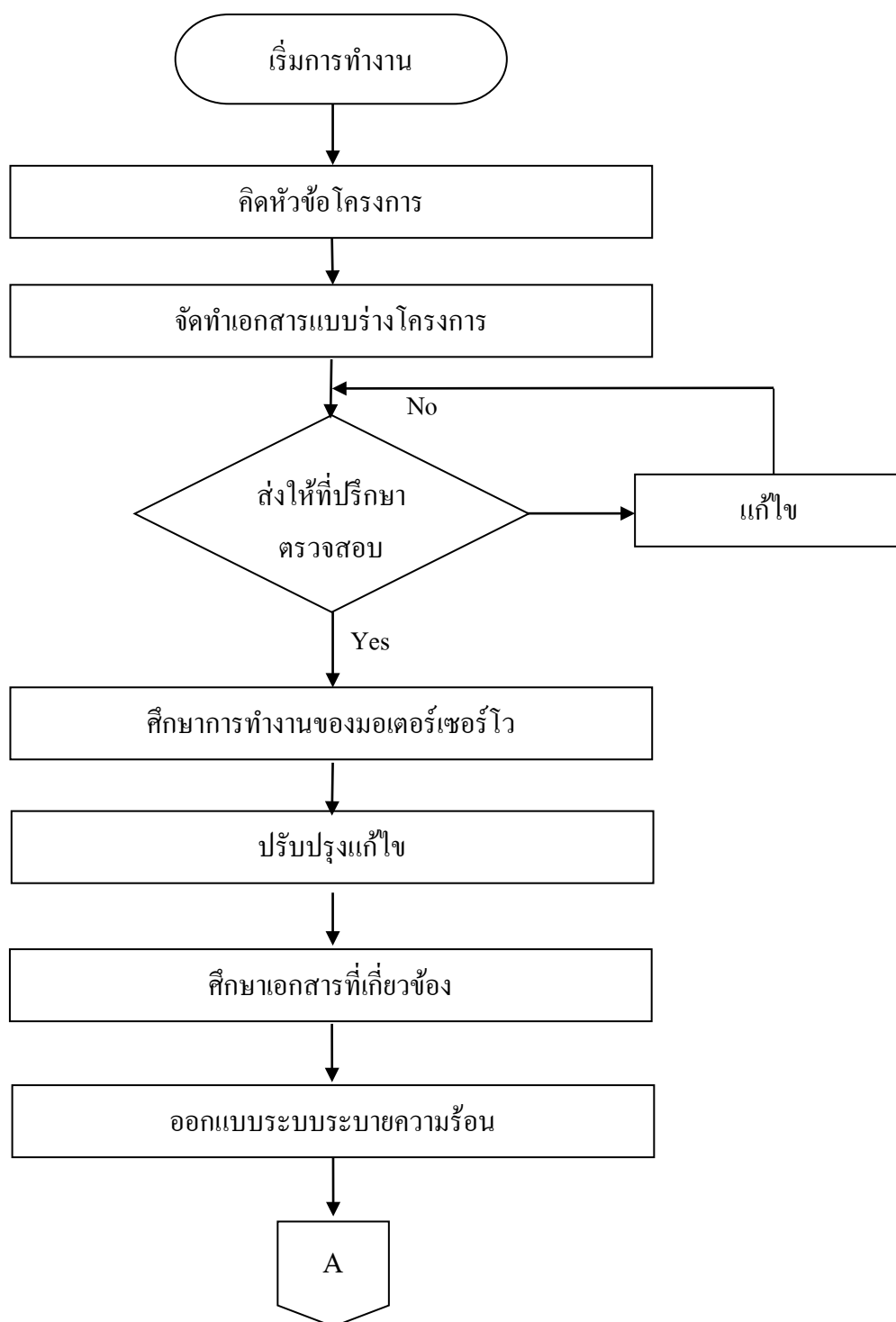
- 1.2.1 เพื่อออกแบบและสร้างระบบควบคุมอุณหภูมิภายในโรงเรือนอัจฉริยะ
- 1.2.2 เพื่อพัฒนาผลงานที่ประดิษฐ์ขึ้นให้เข้าสู่ความเป็นมาตรฐานสามารถนำไปใช้งานได้
อย่างมีคุณภาพประหยัดและปลอดภัย
- 1.2.3 เพื่อให้ผู้จัดทำโครงการนำความรู้จากการศึกษาในหลักสูตรของประกาศนียบัตรวิชาชีพชั้นสูงสาขาวิชาเทคโนโลยีสารสนเทศ มาใช้ในการประกอบอาชีพและการพัฒนา
วิชาชีพของตน

1.3 ขอบเขตของโครงการ

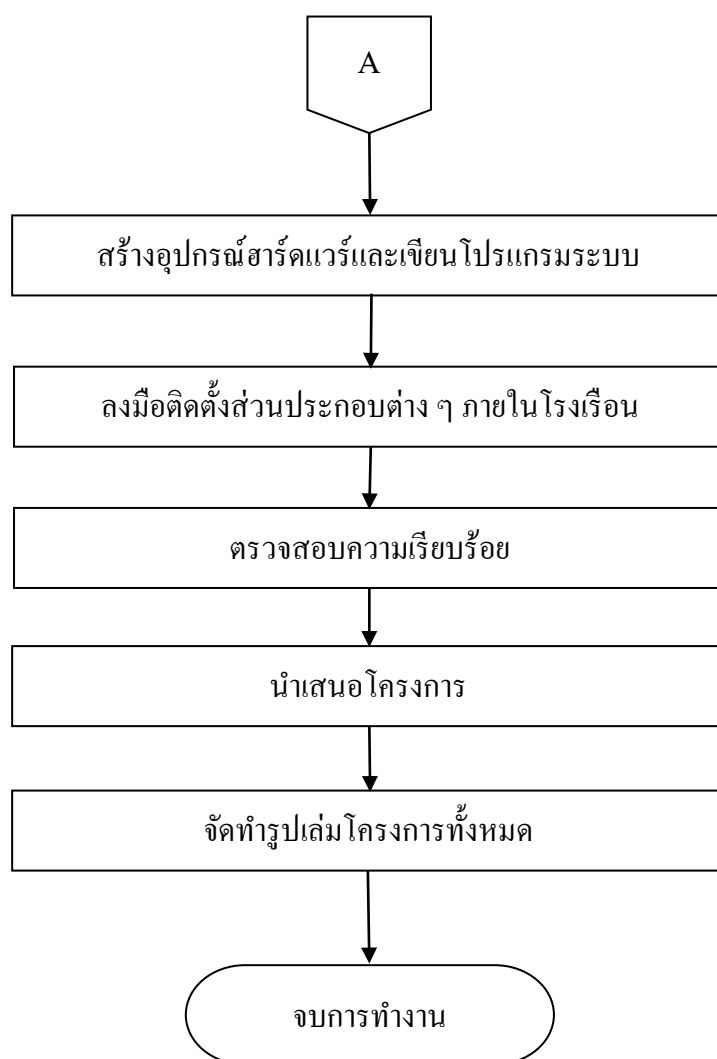
- 1.3.1 สามารถปิด-เปิดการทำงานของหลังคาโรงเรือนได้อัตโนมัติ
- 1.3.2 สามารถปิด-เปิดการทำงานของพัดลมระบายความร้อนโรงเรือนได้อัตโนมัติ
- 1.3.3 สามารถแสดงค่าแสงและค่าอุณหภูมิภายในโรงเรือน

1.4 ขั้นตอนการดำเนินงาน

- 1.4.1 คิดหัวข้อโครงการ
- 1.4.2 จัดทำเอกสารแบบเสนอร่างโครงการ
- 1.4.3 ศึกษาการทำงานของมอเตอร์เซอร์โว
- 1.4.4 ปรับปรุงแก้ไข
- 1.4.5 ศึกษาเอกสารที่เกี่ยวข้อง
- 1.4.6 วิเคราะห์และออกแบบระบบ
- 1.4.7 สร้างอุปกรณ์ฮาร์ดแวร์
- 1.4.8 ทดสอบระบบ
- 1.4.9 ตรวจสอบความเรียบร้อยของระบบ
- 1.4.10 ส่งให้ที่ปรึกษาตรวจสอบ
- 1.4.11 นำเสนอโครงการ
- 1.4.12 จัดทำรูปเล่มโครงการ



รูปที่ 1.1 แผนภาพ Flowchart ขั้นตอนการดำเนินงาน



รูปที่ 1.1 แผนภาพ Flowchart ขั้นตอนการดำเนินงาน (ต่อ)

1.5 ระยะเวลาการทำให้โครงการ

ตารางการดำเนินงานโครงการนี้ใช้ระยะเวลาในการทำ ตั้งแต่เดือน มิถุนายน พ.ศ 2562 ถึง เดือนกุมภาพันธ์ พ.ศ 2563 ดังตารางที่ 1.1

ตารางที่ 1.1 ตารางแผนการดำเนินงาน

ลำดับ	ขั้นตอนการดำเนินงาน	ระยะเวลาการดำเนินงาน								
		ปี พ.ศ.2562 - ปี พ.ศ.2563								
		มิ.ย.	ก.ค.	ส.ค.	ก.ย.	ต.ค.	พ.ย.	ธ.ค.	ม.ค.	ก.พ.
1	คิดหัวข้อโครงการ	←→								
2	จัดทำเอกสารเสนอร่างโครงการ	←→								
3	ศึกษาการทำงานของมอเตอร์เซอร์โว	←→	→							
4	ปรับปรุงแก้ไข		←→							
5	ศึกษาเอกสารที่เกี่ยวข้อง		←→							
6	วิเคราะห์และออกแบบระบบ			←→	→					
7	สร้างอุปกรณ์ฮาร์ดแวร์และติดตั้งระบบ			←→	→	→				
8	ทดสอบระบบ					←→				
9	ตรวจสอบความเรียบร้อยของระบบ					←→				
10	ส่งให้ที่ปรึกษาตรวจสอบ					←→				
11	นำเสนอโครงการ						←→			
12	จัดทำรูปเล่มโครงการ	←	→	→	→	→	→	→	→	→

1.6 ประโยชน์ที่คาดว่าจะได้รับ

- 1.6.1 ระบบระบายความร้อนโรงเรียนเพาะปลูกสามารถทำงานได้อย่างอัตโนมัติ
- 1.6.2 ระบบสามารถแสดงค่าแสงและค่าอุณหภูมิภายในโรงเรียน
- 1.6.3 ผู้จัดทำโครงการนำความรู้จากการศึกษาไปประยุกต์ใช้ในชีวิตประจำวัน

1.7 งบประมาณที่ใช้ในการทำโครงการ

1.7.1 ค่าพิมพ์เอกสาร	200 บาท
1.7.2 ค่าทำเล่มเอกสารโครงการ	200 บาท
1.7.3 ค่ากระดาษ	200 บาท
1.7.4 ค่าอุปกรณ์ในการทำโครงงาน	2,000 บาท
รวม	<u>2,600</u> บาท

บทที่ 2

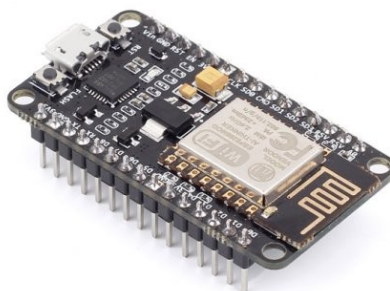
เอกสารและทฤษฎีที่เกี่ยวข้อง

ในการจัดทำโครงการเรื่องระบบรดน้ำต้นไม้อัตโนมัติและแอปพลิเคชันควบคุมโรงเรือนอัจฉริยะ ผู้จัดทำได้ศึกษาเอกสาร เครื่องมือและผลงานที่เกี่ยวข้อง ซึ่งนับว่าเป็นองค์ประกอบที่สำคัญ จะทำให้โครงการมีความครบถ้วนสมบูรณ์มากยิ่งขึ้น คณะผู้จัดทำได้แบ่งข้อมูลที่เกี่ยวข้องเป็นหัวข้อดังนี้

- 2.1 ไมโครคอนโทรลเลอร์ Node MCU
- 2.2 บอร์ด Arduino
- 2.3 LCD Display
- 2.4 Relay 4 Chanel
- 2.5 BH1750 Lux Sensor
- 2.6 Servo Motor
- 2.7 Temperature Sensor
- 2.8 หลักการเขียนโปรแกรม Arduino

2.1 ไมโครคอนโทรลเลอร์ Node MCU

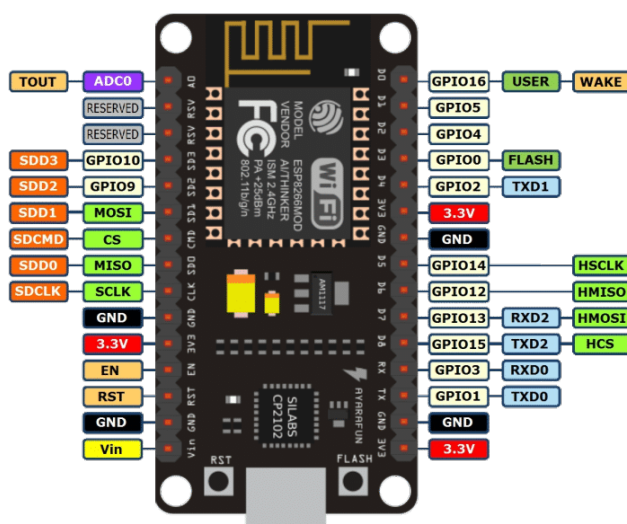
NodeMCU คือ แพลตฟอร์มหนึ่งที่ใช้ช่วยในการสร้างโปรเจกอินเทอร์เนตของสรรพสิ่ง (Internet of Things, IoT) ที่ประกอบไปด้วยตัวบอร์ดและซอฟต์แวร์ในตัวFirmware ที่เป็นลักษณะโอเพ่นซอร์ส(open-source) ในตอนแรกผู้ใช้งานจะต้องเขียนโปรแกรมด้วยภาษา Lua เพื่อให้ใช้งาน NodeMCU ได้ แต่ต่อมาได้พัฒนาให้สามารถใช้ภาษา C ในArduino IDE เพื่อควบคุมได้ใช้งานได้ง่ายขึ้นทำให้บอร์ดนี้มาพร้อมกับโมดูลWIFI (ESP8266) ซึ่งเป็นหัวใจสำคัญในการใช้เชื่อมต่อกับอินเทอร์เน็ต ตัวโมดูลWiFi (ESP8266) นั้นมีอยู่ด้วยกันหลายรุ่น ตั้งแต่เวอร์ชันแรกคือ ESP-01 จนถึงเวอร์ชัน ESP-12 โดยโมดูลนี้ที่เป็นองค์ประกอบใน NodeMCU รุ่นแรกนั้นคือ ESP-12 และในเวอร์ชันต่อมา NodeMCU(v.2) นั้นจะใช้โมดูล ESP-12E แทน ซึ่งในการใช้งานโดยรวมนั้นจะไม่แตกต่างกันมากนัก แต่บอร์ด NodeMCU v.3 จะมีขนาดใหญ่ขึ้น NodeMCU นั้นมีลักษณะคล้ายกับ Arduino ตรงที่มีพอร์ตขาเข้าหรืออินพุตและพอร์ตขาออกหรือเอาต์พุตพอร์ตอยู่ในตัวเอง ทำให้เราสามารถเขียนโปรแกรมควบคุมอุปกรณ์ต่าง ๆ ได้โดยตรงไม่ต้องทำผ่านชิปอื่น ๆ ซึ่ง NodeMCU v.3 นี้สามารถทำอะไรได้หลายอย่างมาก โดยเฉพาะงานที่เกี่ยวข้องกับ IoT ไม่ว่าจะเป็นการทำเว็บเซิร์ฟเวอร์ขนาดเล็ก การควบคุมการเปิดปิดไฟผ่านเครือข่าย และอื่น ๆ อีกมากมาย



รูปที่ 2.1 ภาพบอร์ด NodeMCU v.2

ที่มา : (<https://inex.co.th/shop/media/catalog/product/cache/1/image/n/o/node-mcu-v2-001.jpg>)

NodeMCU Development Kit V2 เป็นตัวที่พัฒนาจาก NodeMCU Version เดิม โดยเป็นโมดูลที่ประกอบด้วย ESP8266-12 E มีเสาอากาศแบบ PCB Antenna เชื่อมต่อเสดเคอร์สำหรับขาสัญญาณต่าง ๆ ได้แก่ GPIO, PWM, I2C, 1-wire, ADC และ มี SPI เพิ่มขึ้นมาจาก Version เดิม มีส่วนของ USB-to-TTL และพอร์ต micro USB ซึ่งใช้ชิพ USB to Serial ของ silicon labcp 2102 เชื่อมต่อเข้ากับเครื่องคอมพิวเตอร์เพื่อพัฒนาโปรแกรม สามารถติดตั้งเฟิร์มแวร์ NodeMCU ได้ และยังมีขนาดของ PCB ที่เล็กลง สามารถใช้งานกับ breadboard ได้ ผู้ใช้สามารถเลือกพัฒนาด้วยสคริปต์ Lua โดยใช้เฟิร์มแวร์ NodeMCU หรือใช้เป็นชุดพัฒนาด้วยโมดูล ESP8266 ก็ได้ซึ่งสามารถเขียนด้วย Arduino IDE ได้ โมดูลมี GPIO ให้ใช้ถึง 10 พอร์ต สามารถนำมาพัฒนาโปรเจกต์ทางด้าน Internet of Things (IoTs) เชื่อมต่อกับอุปกรณ์อื่น ๆ ตามต้องการ



รูปที่ 2.2 แสดงตำแหน่งพินและขาของ NodeMCU v.2

ที่มา : (https://i.lnwfile.com/_/i/_raw/5c/2x/x5.jpg)

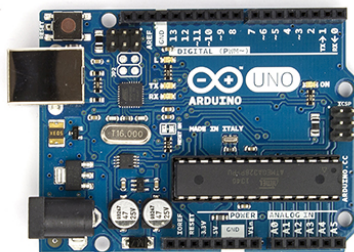
ข้อมูลสำคัญเชิงเทคนิคของบอร์ด NodeMCU v2

- ใช้โมดูล ESP-12E (ESP8266 SoC chip) ของบริษัท AI Thinker (ในขณะที่ NodeMCU v1 ใช้โมดูล ESP12) มีขาเพิ่มมาอีก 6 ขา เมื่อเปรียบเทียบกับ ESP-12
- ใช้ชิป Flash ความจุ 32Mbits (4MBytes)
- มีขนาดแคบกว่า NodeMCU v1 ดังนั้นเมื่อเสียบขาลงบนเบรคบอร์ด จะมีช่องเหลือด้านข้าง ทำให้สะดวกในการต่อวงจรบนเบรคบอร์ด
- มีวงจรควบคุมแรงดัน 3.3V (@800mA max.) บนบอร์ด ใช้ไอซีที่จ่ายกระแสได้มากกว่าบอร์ด NodeMCU v1
- ใช้ชิป CP2102 ของ Silabs ทำหน้าที่เป็นส่วนเชื่อมต่อ USB-to-Serial (แต่ NodeMCU v1 ใช้ชิป CH340G)
- มีขาสำหรับ SPI สำหรับต่อกับการ์ด SD (เพิ่มจากเดิมที่มีขาสำหรับ HSPI)
- มีขา GPIO3/RXD0 และ GPIO1/TXD0 ที่ต่อกับขา TXD และ RXD ของชิป CP2102 ตามลำดับ
- มีขา GPIO13/RXD2 และ GPIO15/TXD2 (ใช้เป็นพอร์ต Serial เพิ่มอีกหนึ่งชุด)
- ใช้คอนเนกเตอร์แบบ micro-USB สำหรับจ่ายแรงดันไฟเลี้ยง (VUSB) เท่ากับ +5V
- สามารถจ่ายแรงดันไฟเลี้ยง +5V จากภายนอกได้ (ต่อเข้าที่ขา VDD5V)
- มีปุ่มกด RST (รีเซ็ตการทำงาน) และ Flash (สำหรับโปรแกรมเฟิร์มแวร์ใหม่)
- มีขา A0 รับอินพุตแรงดันแบบแอนะล็อกสำหรับวงจร ADC (ขนาด 10 บิต) ที่อยู่ในชิป ผ่านวงจรแบ่งแรงดันด้วยตัวต้านทาน 100k / 220k (ลดแรงดันอินพุตจาก 0..3.3V ลงมาให้อยู่ในช่วง 0V-1V)

2.2 บอร์ด Arduino

อาดูอินเป็นภาษาอิตาลี ซึ่งใช้ในการเรียกโครงการพัฒนาไมโครคอนโทรลเลอร์ตระกูล AVR แบบ Open Source โดยจุดประสงค์เพื่อให้ผู้เริ่มต้นพัฒนาสร้างบอร์ดและวงจรขนาดเล็ก ต้นทุนต่ำ เพื่อพัฒนาวัตถุตอบโต้แบบพึ่งพาตนเอง (Standalone) สามารถพัฒนาโปรแกรมผ่านระบบได้หลากหลายระบบปฏิบัติการได้อย่างง่ายดาย (Banzi & Shiloh, 2014) จากภาพที่ 2.3 Arduino RS232 เป็นต้นแบบที่ถูกพัฒนาขึ้นในปี 2004 ถูกพัฒนาโดยนาย Massimo Banzi จากการที่นักเรียนนักศึกษาที่เรียนเกี่ยวกับวิชาอิเล็กทรอนิกส์ในเมืองไอวีเรีย (Ivrea) ประเทศอิตาลี ต้องซื้อไมโครคอนโทรลเลอร์เบสิก (BASIC Stamp) ที่มีราคาสูงถึง 100 ดอลลาร์สหรัฐ ซึ่งนาย Massimo Banzi เห็นว่านักเรียนต้องซื้อสินค้าราคาแพง ราคาไม่สมเหตุสมผล จึงปรึกษากับนาย Hernando

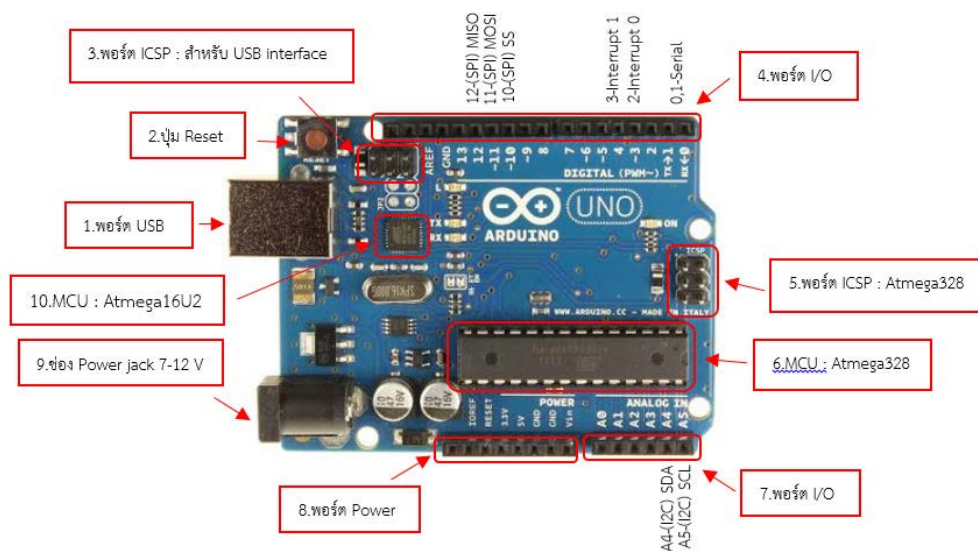
Barragan ให้ช่วยออกแบบวงจรขึ้นมา และนอกจากนี้ นาย David Cuartielles ได้ แนะนำว่าควรให้บอร์ดนี้เป็นระบบโอเพนซอร์ซ (Open Source) เพื่อให้ทุกคนสามารถเข้าถึงได้



รูปที่ 2.3 Arduino RS232 ต้นแบบของอาคุยโน

ที่มา : (https://i.lnwfile.com/_/i/_raw/43/0h/cu.jpg)

อาคุยโนใช้พื้นฐานของภาษาซีพลัสพลัส (C++) เป็นรูปแบบของโปรแกรมภาษาซีประยุกต์ มีโครงสร้างของตัวภาษาโดยรวมใกล้เคียงกับภาษาซีมาตรฐาน (ANSI-C) นักพัฒนาไม่จำเป็นต้องเขียนการควบคุมเองทั้งหมด สามารถเลือกจากคำสั่งจากโปรแกรมแปลภาษาซี (C-Compiler) ทำให้การพัฒนาเป็นไปได้ง่ายมากที่สุด



รูปที่ 2.4 Layout & Pin out Arduino Board (Model: Arduino UNO R3)

ที่มา : (<https://www.thaieasyelec.com/media/wysiwyg/r7.jpg>)

จากภาพที่ 2.4 ส่วนประกอบพื้นของอาคยโนจะประกอบด้วยทั้งหมด 10 ส่วนด้วยกัน ได้แก่

- 1) USBPort: ใช้สำหรับต่อกับ Computer เพื่ออัปโหลดโปรแกรมเข้า MCU และจ่ายไฟให้กับบอร์ด
- 2) Reset Button: เป็นปุ่ม Reset ใช้กดเมื่อต้องการให้ MCU เริ่มการทำงานใหม่
- 3) ICSP Port ของ Atmega16U2 เป็นพอร์ตที่ใช้โปรแกรม Visual Com port บน Atmega16U2
- 4) I/O Port: Digital I/O ตั้งแต่ขา D0 ถึง D13 นอกจากนี้ บาง Pin จะทำหน้าที่อื่น ๆ เพิ่มเติมด้วย เช่น Pin0,1 เป็นขา Tx,Rx Serial, Pin3,5,6,9,10 และ 11 เป็นขา PWM
- 5) ICSP Port: Atmega328 เป็นพอร์ตที่ใช้โปรแกรม Bootloader
- 6) MCU: Atmega328 เป็น MCU ที่ใช้บนบอร์ด Arduino
- 7) I/O Port: นอกจากจะเป็น Digital I/O แล้ว ยังเปลี่ยนเป็น ช่องรับสัญญาณอนาล็อกตั้งแต่ขา A0-A5
- 8) Power Port: ไฟเลี้ยงของบอร์ดเมื่อต้องการจ่ายไฟให้กับวงจรภายนอก ประกอบด้วยไฟเลี้ยง +3.3 V, +5V, GND, Vin
- 9) Power Jack: รับไฟจาก Adapter โดยที่แรงดันอยู่ระหว่าง 7-12 V
- 10) MCU ของ Atmega16U2 เป็น MCU ที่ทำหน้าที่เป็น USB to Serial โดย Atmega328 จะติดต่อกับ Computer ผ่าน Atmega16U2

2.3 LCD Display (Liquid Crystal Display)



รูปที่ 2.5 แสดงตัวอย่าง LCD

ที่มา : (<https://inwfile.com/s-i/uc3ixf.jpg>)

คำว่า LCD ย่อมาจากคำว่า Liquid Crystal Display ซึ่งเป็นจอที่ทำมาจากผลึกคริสตัลเหลว หลักการคือด้านหลังจอจะมีไฟส่องสว่าง หรือที่เรียกว่า Backlight อยู่ เมื่อมีการปล่อยกระแสไฟฟ้าเข้าไปกระตุ้นที่ผลึก ก็จะทำให้ผลึกโปร่งแสง ทำให้แสงที่มาจากไฟ Backlight แสดงขึ้นมบบนหน้าจอ ส่วนอื่นที่โดนผลึกปิดกั้นไว้ จะมีสีที่แตกต่างกันตามสีของผลึกคริสตัล เช่น สีเขียว หรือ สีฟ้า ทำให้เมื่อมองไปที่จอก็จะพบกับตัวหนังสือสีขาว แล้วพบกับพื้นหลังสีต่าง ๆ กัน

จอ LCD จะแบ่งเป็น 2 แบบใหญ่ๆตามลักษณะการแสดงผลดังนี้

- Character LCD เป็นจอที่แสดงผลเป็นตัวอักษรตามช่องแบบตายตัว เช่น จอ LCD ขนาด 16x2 หมายถึงใน 1 แถว มีตัวอักษรใส่ได้ 16 ตัว และมีทั้งหมด 2 บรรทัดให้ใช้งาน ส่วน 20x4 จะหมายถึงใน 1 แถว มีตัวอักษรใส่ได้ 20 ตัว และมีทั้งหมด 2 บรรทัด

- Graphic LCD เป็นจอที่สามารถกำหนดได้ว่าจะให้แต่ละจุดบนหน้าจอขึ้นแสง หรือปล่อยแสงออกไป ทำให้อจอสามารถสร้างรูปขึ้นมาบนหน้าจอได้ การระบุขนาดจะระบุในลักษณะของจำนวนจุด (Pixels) ในแต่ละแนว เช่น 128x64 หมายถึงจอที่มีจำนวนจุดตามแนวนอน 128 จุด และมีจุดตามแนวตั้ง 64 จุด

2.3.1 การเชื่อมต่อกับจอ Character LCD

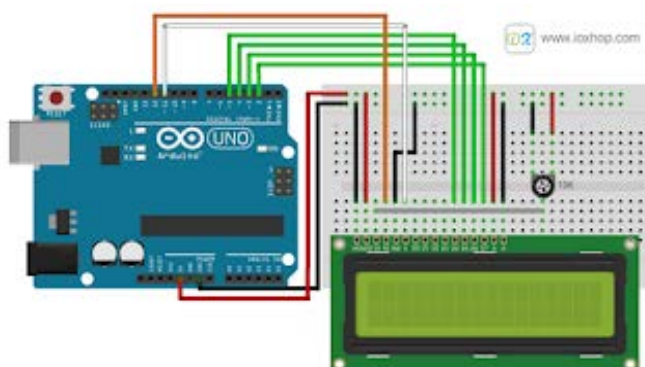
การเชื่อมต่อจะมีด้วยกัน 2 แบบ คือ

2.3.1.1 การเชื่อมต่อแบบขนาน - เป็นการเชื่อมต่อจอ LCD เข้ากับบอร์ด Arduino โดยตรง โดยจะแบ่งเป็นการเชื่อมต่อแบบ 4 บิต และการเชื่อมต่อแบบ 8 บิต ใน Arduino จะนิยมเชื่อมต่อแบบ 4 บิต เนื่องจากใช้สายในการเชื่อมต่อน้อยกว่า

2.3.1.2 การเชื่อมต่อแบบอนุกรม - เป็นการเชื่อมต่อกับจอ LCD ผ่าน โมดูลแปลงรูปแบบการเชื่อมต่อกับจอ LCD จากแบบขนาน มาเป็นการเชื่อมต่อแบบอื่นที่ใช้สายน้อยกว่า เช่น การใช้โมดูล I2C Serial Interface จะเป็นการนำโมดูลเชื่อมเข้ากับตัวจอ LCD แล้วใช้บอร์ด Arduino เชื่อมต่อกับบอร์ดโมดูลผ่านโปรโตคอล I2C ทำให้ใช้สายเพียง 4 เส้น ก็ทำให้หน้าจอแสดงผลข้อความต่าง ๆ ออกมาได้

2.3.2 การใช้งาน Character LCD กับ Arduino และการเชื่อมต่อแบบขนาน

การเชื่อมต่อแบบขนานแบบ 4 บิต สามารถทำได้ตามวงจรด้านล่างนี้



รูปที่ 2.6 แสดงการต่อ LCD แบบขนาน

ที่มา : (<https://www.ioxhop.com/article/30/>)

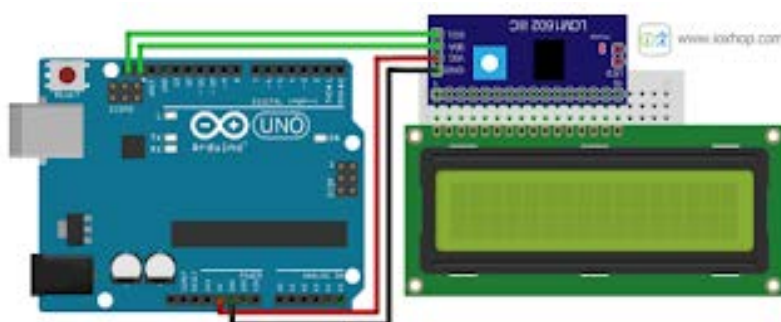
เมื่อต่อวงจรเรียบร้อยแล้ว ต่อสาย USB เข้ากับบอร์ด Arduino จะเห็นกล่องสี่เหลี่ยมทั้งหมด 16 ตัว (หากเป็นจอ 16x2) ในบรรทัดแรก หากไม่พบกล่อง ให้ปรับความชัดได้จาก VR ที่ต่ออยู่กับขา V0

2.3.3 การควบคุมไฟแบล็คไลท์

ขั้วขา A ที่ต่ออยู่กับขั้วบวก มาต่อที่ขา Digital Pin แทนจากนั้นใช้คำสั่ง pinMode() และ digitalWrite() สั่งเปิด-ปิดไฟแบล็คไลท์ได้แบบเดียวกับการควบคุมการติดดับของหลอด LED

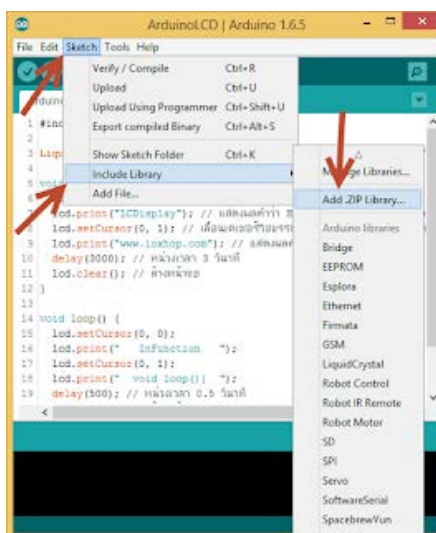
2.3.4 การเชื่อมต่อแบบอนุกรม (LCD I2C)

การเชื่อมต่อแบบอนุกรม จะใช้งานโมดูล I2C Serial Interface Board Module มาเชื่อมต่อระหว่าง Arduino กับจอ LCD วงจรที่เชื่อมต่อจะเป็นไปตามรูปนี้ (กรณีใช้บอร์ดรุ่นอื่น จะต้องต่อ SDA เข้า A4 และ SCL เข้ากับ A5)



รูปที่ 2.7 แสดงการต่อ LCD แบบอนุกรม

ที่มา : (<https://www.ioxhop.com/article/30/>)



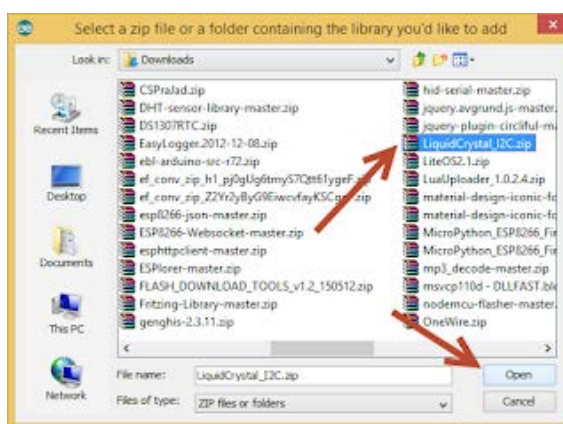
ดาวน์โหลด Library ได้จาก : LiquidCrystal_I2C.zip แล้วเพิ่มไลบรารีตามขั้นตอนต่อไปนี้

รูปที่ 2.8 แสดงการเพิ่ม Library ในโปรแกรม Arduino IDE

ที่มา : (<https://www.ioxhop.com/article/30/AD-character-lcd>)

2.3.4.1 เปิดโปรแกรม Arduino IDE ขึ้นมา จากนั้นกดไปที่ Tool > Include Library

> Add .ZIP Library



รูปที่ 2.9 แสดงไลบรารีได้ถูกเพิ่มเข้ามาแล้ว

ที่มา : (<https://www.ioxhop.com/article/30/AD-character-lcd>)

2.3.4.2 เลือกไฟล์ที่ได้ดาวน์โหลดไว้ในขั้นตอนที่แล้ว จากนั้นกดปุ่ม Open Library ได้ถูกเพิ่มเข้ามาแล้ว นำโค้ดต่อไปนี้ อัปโหลดลงบอร์ด Arduino

2.3.4.3 Library ได้ถูกเพิ่มเข้ามาแล้ว นำโค้ดต่อไปนี้อัปโหลดลงบอร์ด Arduino

```

#include <Wire.h>
#include <LiquidCrystal_I2C.h>

// Set the LCD address to 0x27 in PCF8574 by NXP and Set to 0x3F in PCF8574A by Ti
LiquidCrystal_I2C lcd( 0x3F, 16, 2); // จอกว้าง 16 ตัวอักษร 2 บรรทัด รหัสประจำตัว 0x3F

void setup() {
    lcd.begin();
    lcd.print("LCDisplay"); // แสดงผลคำว่า Hello, world! ออกหน้าจอ
    lcd.setCursor(0, 1); // เลื่อนเคเซอร์ไปบรรทัดที่ 2 ลำดับที่ 0 (ก่อนหน้าตัวอักษรแรก)
    lcd.print("www.ioxhop.com"); // แสดงผลคำว่า www.ioxhop.com
    delay(3000); // หน่วงเวลา 3 วินาที
    lcd.clear(); // ล้างหน้าจอ
}

void loop() {
    lcd.setCursor(3, 0);
    lcd.print("InFunction ");
    lcd.setCursor(2, 1);
    lcd.print("void loop(){ ");
    delay(500); // หน่วงเวลา 0.5 วินาที
    lcd.clear(); // ล้างหน้าจอ
    delay(500); // หน่วงเวลา 0.5 วินาที
}

```

ตรง 0x3F หากอัปโหลดแล้วไม่สามารถใช้งานได้ (ไม่มีอะไรแสดงผลออกทางหน้าจอ)
ลองแก้เป็น 0x27 แล้วอัปโหลดเข้าไปใหม่อีกครั้ง



รูปที่ 2.10 การเปลี่ยนหมายเลขประจำตัว (I2C Address)

ที่มา : (<http://www.thaimicrotron.com/CCS-628/Reference/I2CBUS.html>)

ในกรณีที่ต้องการต่อจอหลายๆจอโดยใช้ไมโครตัวเดียวกัน สามารถเปลี่ยนหมายเลขประจำตัว หรือ Address ได้จากการจัมแนวยาว A0 หรือ A1 หรือ A2 ทำให้ A0 หรือ A1 หรือ A2 มีลอจิกเป็น 0 ทำให้หมายเลขประจำตัวเปลี่ยนไปดังตารางด้านล่างนี้

- กรณีชิปบนโมดูลเป็นเบอร์ PCF8574

ตารางที่ 2.1 แสดงค่าโมดูลเบอร์ PCF8574

Table 4. PCF8574 address map

Pin connectivity			Address of PCF8574								Address byte value		7-bit hexadecimal address without R/W
A2	A1	A0	A6	A5	A4	A3	A2	A1	A0	R/W	Write	Read	
V _{SS}	V _{SS}	V _{SS}	0	1	0	0	0	0	0	-	40h	41h	20h
V _{SS}	V _{SS}	V _{DD}	0	1	0	0	0	0	1	-	42h	43h	21h
V _{SS}	V _{DD}	V _{SS}	0	1	0	0	0	1	0	-	44h	45h	22h
V _{SS}	V _{DD}	V _{DD}	0	1	0	0	0	1	1	-	46h	47h	23h
V _{DD}	V _{SS}	V _{SS}	0	1	0	0	1	0	0	-	48h	49h	24h
V _{DD}	V _{SS}	V _{DD}	0	1	0	0	1	0	1	-	4Ah	4Bh	25h
V _{DD}	V _{DD}	V _{SS}	0	1	0	0	1	1	0	-	4Ch	4Dh	26h
V _{DD}	V _{DD}	V _{DD}	0	1	0	0	1	1	1	-	4Eh	4Fh	27h

- กรณีชิปบนโมดูลเป็นเบอร์ PCF8574A

ตารางที่ 2.2 แสดงโมดูลเบอร์ PCF8574A

Table 5. PCF8574A address map

Pin connectivity			Address of PCF8574A								Address byte value		7-bit hexadecimal address without R/W
A2	A1	A0	A6	A5	A4	A3	A2	A1	A0	R/W	Write	Read	
V _{SS}	V _{SS}	V _{SS}	0	1	1	1	0	0	0	-	70h	71h	38h
V _{SS}	V _{SS}	V _{DD}	0	1	1	1	0	0	1	-	72h	73h	39h
V _{SS}	V _{DD}	V _{SS}	0	1	1	1	0	1	0	-	74h	75h	3Ah
V _{SS}	V _{DD}	V _{DD}	0	1	1	1	0	1	1	-	76h	77h	3Bh
V _{DD}	V _{SS}	V _{SS}	0	1	1	1	1	0	0	-	78h	79h	3Ch
V _{DD}	V _{SS}	V _{DD}	0	1	1	1	1	0	1	-	7Ah	7Bh	3Dh
V _{DD}	V _{DD}	V _{SS}	0	1	1	1	1	1	0	-	7Ch	7Dh	3Eh
V _{DD}	V _{DD}	V _{DD}	0	1	1	1	1	1	1	-	7Eh	7Fh	3Fh

2.3.5 ฟังก์ชันสั่งงานจอ LCD

`lcd.clear()` > ใช้ล้างหน้าจอ เมื่อมีตัวอักษรใดๆอยู่บนหน้าจอ จะถูกล้างออกทั้งหมด

`lcd.home()` > ใช้ปรับให้เคเซอร์กลับไปอยู่ที่ตำแหน่งแรกด้านซ้าย เมื่อใช้คำสั่ง `lcd.print()` จะไปเริ่มแสดงผลทางด้านบนซ้าย

`lcd.setCursor()` (ลำดับตัวอักษรนับจากทางซ้าย, บรรทัด) > ใช้ตั้งค่าเคเซอร์ เช่น `lcd.setCursor(2,0);` หมายถึงเซตเคเซอร์ไปตัวอักษรที่ 2 นับจากทางซ้ายและอยู่บรรทัดแรกเมื่อใช้คำสั่ง `lcd.print()` ตัวอักษรตัวแรกจะอยู่ลำดับที่ 3 นับจากทางซ้าย

`lcd.write` (ข้อมูลที่ต้องการเขียนออกไป) > ใช้สำหรับเขียนข้อมูลออกไปทีละตัวอักษร
`lcd.print` (ข้อมูลที่ต้องการให้เขียนออกไป (รูปแบบข้อมูล) > ใช้เขียนข้อมูลออกไปทั้งข้อความ
`lcd.cursor()` > ใช้สั่งให้แสดงเคเซอร์บนหน้าจอ
`lcd.noCursor()` > ใช้สั่งให้ไม่แสดงเคเซอร์บนหน้าจอ
`lcd.display()` > แสดงตัวอักษรบนหน้าจอ
`lcd.noDisplay()` > ปิดการแสดงตัวอักษรในหน้าจอ
`lcd.scrollDisplayLeft()` > เลื่อนตัวอักษรไปทางซ้าย 1 ตัว
`lcd.scrollDisplayRight()` > เลื่อนตัวอักษรไปทางขวา 1 ตัว
`lcd.autoscroll()` > เลื่อนตัวอักษรไปทางขวาอัตโนมัติหากใช้คำสั่ง `lcd.print()` หรือ `lcd.write()` เมื่อตัวอักษรเต็มหน้าจอ
`lcd.noAutoscroll()` > ปิดการเลื่อนตัวอักษรอัตโนมัติ
`lcd.leftToRight()` > เมื่อใช้คำสั่ง `lcd.print()` หรือ `lcd.write()` ตัวอักษรจะเขียนจากซ้ายไปขวา
`lcd.rightToLeft()` > เมื่อใช้คำสั่ง `lcd.print()` หรือ `lcd.write()` ตัวอักษรจะเขียนจากขวาไปซ้าย

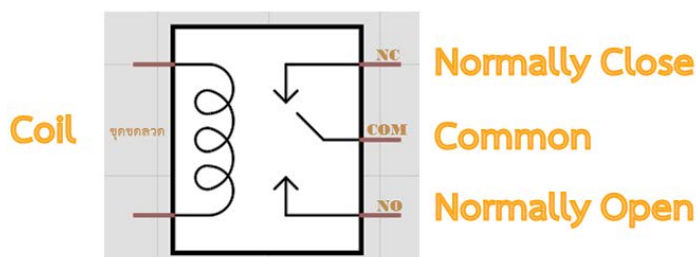
2.4 Relay 4 Chanel



รูปที่ 2.11 แสดงตัวอย่าง Relay

ที่ ม ๑ : (<https://www.thaieasyelec.com/article-wiki/review-product-article/Example-project-for-control-electrical-device-using-arduino-and-relay-module.html>)

รีเลย์ (Relay) เป็นอุปกรณ์ไฟฟ้าชนิดหนึ่ง ซึ่งทำหน้าที่ตัดต่อวงจรแบบเดียวกับสวิตช์ โดยควบคุมการทำงานด้วยไฟฟ้า Relay มีหลายประเภท ตั้งแต่ Relay ขนาดเล็กที่ใช้ในงานอิเล็กทรอนิกส์ทั่วไป จนถึง Relay ขนาดใหญ่ที่ใช้ในงานไฟฟ้าแรงสูง โดยมีรูปร่างหน้าตาแตกต่างกันออกไป แต่มีหลักการทำงานที่คล้ายคลึงกัน สำหรับการนำ Relay ไปใช้งาน จะใช้ในการตัดต่อวงจร ทั้งนี้ Relay ยังสามารถเลือกใช้งานได้หลากหลายรูปแบบ



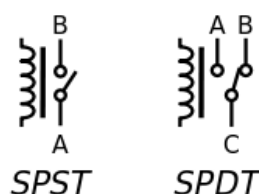
รูปที่ 2.12 สัญลักษณ์ในวงจรไฟฟ้าของรีเลย์

ที่มา : (<http://www.atom.rmutphysics.com/charud/oldnews/0/286/4/11/relay/relay.htm>)

ภายใน Relay จะประกอบไปด้วยขดลวดและหน้าสัมผัส

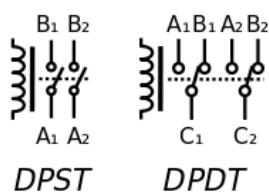
- หน้าสัมผัส NC (Normally Close) เป็นหน้าสัมผัสปกติปิด โดยในสภาวะปกติหน้าสัมผัสนี้จะต่อเข้ากับขา COM (Common) และจะลดยหรือไม่สัมผัสกันเมื่อมีกระแสไฟฟ้าไหลผ่านขดลวด
- หน้าสัมผัส NO (Normally Open) เป็นหน้าสัมผัสปกติเปิด โดยในสภาวะปกติจะลดยอยู่ไม่ถูกต่อกับขา COM (Common) แต่จะเชื่อมต่อกันเมื่อมีกระแสไฟฟ้าไหลผ่านขดลวด
- ขา COM (Common) เป็นขาที่ถูกใช้งานร่วมกันระหว่าง NC และ NO ขึ้นอยู่กับว่าขณะนั้นมีกระแสไฟฟ้าไหลผ่านขดลวดหรือไม่ หน้าสัมผัสใน Relay 1 ตัวอาจมีมากกว่า 1 ชุด ขึ้นอยู่กับผู้ผลิตและลักษณะของงานที่ถูกนำไปใช้ จำนวนหน้าสัมผัสถูกแบ่งออกดังนี้

สวิตช์จะถูกแยกประเภทตามจำนวน Pole และจำนวน Throw ซึ่งจำนวน Pole (SP-Single Pole, DP-Double Pole, 3P-Triple Pole, etc.) จะบอกถึงจำนวนวงจรที่ทำการเปิด-ปิด หรือ จำนวนของขา COM นั้นเอง และจำนวน Throw (ST, DT) จะบอกถึงจำนวนของตัวเลือกของ Pole ตัวอย่างเช่น SPST- Single Pole Single Throw สวิตช์จะสามารถเลือกได้เพียงอย่างเดียวโดยจะเป็นปกติเปิด (NO-Normally Open) หรือปกติปิด (NC-Normally Close) แต่ถ้าเป็น SPDT- Single Pole Double Throw สวิตช์จะมีหนึ่งคู่เป็นปกติเปิด (NO) และอีกหนึ่งคู่เป็นปกติปิดเสมอ(NC) ดังรูป



รูปที่ 2.13 แสดงตัวอย่าง SPST และ SPDT

ที่มา : (<https://sites.google.com/site/cududegroup/electrical/othercomponent>)



รูปที่ 2.14 แสดงตัวอย่าง DPST และ DPDT

ที่มา : (<https://sites.google.com/site/cududegroup/electrical/othercomponent>)

2.4.1 Relay Module 4 Channels



รูปที่ 2.15 แสดงตัวอย่าง Relay Module 4 Channels

ที่มา : ([http://www.mltelectronic.com/-5V-Optocoupler-\(4-Channel-Relay-Module\)](http://www.mltelectronic.com/-5V-Optocoupler-(4-Channel-Relay-Module)))

Relay Module 4 Channels มีเอาต์พุตคอนเน็คเตอร์ที่ Relay เป็น NO/COM/NC สามารถใช้กับโหลดได้ทั้งแรงดันไฟฟ้า DC และ AC โดยใช้สัญญาณในการควบคุมการทำงานด้วยสัญญาณลอจิก TTL

คุณสมบัติ (Features)

- รีเลย์เอาต์พุตแบบ SPDT จำนวน 4 ช่อง
- ทำงานด้วยระดับแรงดัน TTL
- CONTACT OUTPUT ของรีเลย์รับแรงดันได้สูงสุด 250 VAC 10 A, 30 VDC 10 A
- มี LED แสดงสถานะ การทำงานของรีเลย์และแสดงสถานะของบอร์ด
- มีจัมป์เปอร์สำหรับเลือกว่าจะใช้กราวด์ร่วมหรือแยก
- มี OPTO-ISOLATED เพื่อแยกกราวด์ส่วนของสัญญาณควบคุมกับไฟฟ้าที่ขับรีเลย์ออกจากกัน

ตารางที่ 2.3 แสดงขาที่ใช้ในการเชื่อมต่อของ Relay Module 4 Channels

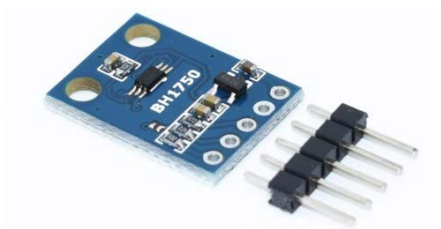
ขาที่	คำอธิบาย
1	+VCC ขาไฟ 5VDC
2	GND
3	ขาสัญญาณอินพุต Relay 1 (IN1)
4	ขาสัญญาณอินพุต Relay 2 (IN2)
5	ขาสัญญาณอินพุต Relay 3 (IN3)
6	ขาสัญญาณอินพุต Relay 4 (IN4)
7	COM (คอมมอนของ OPTO)
8	GND (กราวด์ของบอร์ดเป็นกราวด์เดียวกันกับขาที่ 2)
9	NC (Normal Close) ซึ่งหมายถึงหน้าสัมผัสแบบปกติปิด
10	COM (Common) ที่จะตัดหรือต่อวงจรจากขา NC, NO
11	NO (Normal Open) ซึ่งหมายถึงหน้าสัมผัสแบบปกติเปิด

2.5 BH1750 Lux Sensor

BH1750 Lux Sensor คือ เซ็นเซอร์วัดแสงที่ให้เอาต์พุตออกมาเป็นสัญญาณดิจิทัล และเอาต์พุตที่ได้ก็มีหน่วยเป็นลักซ์ เราจึงไม่จำเป็นต้องใช้การคำนวณเอาต์พุตเพิ่มเติมใดๆ เซ็นเซอร์ BH1750FVI จึงเป็นข้อได้เปรียบหรือเรียกว่ามีข้อดีกว่าเซ็นเซอร์วัดแสงจำพวกโฟโตทรานซิสเตอร์หรือโฟโตไดโอด เนื่องจากเซนเซอร์ดังกล่าวให้เอาต์พุตเป็นการเปลี่ยนแปลงแรงดัน หรือ เซ็นเซอร์ LDR ที่ให้ให้เอาต์พุตเป็นการเปลี่ยนแปลงค่าความต้านทาน

2.5.1 คุณสมบัติทางเทคนิค

- ใช้ไอซี BH1750FVI ของบริษัท ROHM SEMICONDUCTOR
- สามารถวัดความเข้มแสงได้ตั้ง แต่ 1-65535 lx ความละเอียด 16 บิต
- ใช้ไฟเลี้ยง 3V-5V
- ใช้การเชื่อมต่อแบบ I2C ความถี่สูงสุด 400 kHz
- มีขา ADD สำหรับเลือก Address ของบอร์ดได้ 2 ค่า
- เชื่อมต่อสัญญาณโดยตรงกับไมโครคอนโทรลเลอร์ระบบไฟ 3.3V และระบบไฟ 5V ได้ทันทีโดยไม่ต้องผ่านวงจรปรับระดับแรงดัน
- ขนาด PCB 33 mm x 15.2 mm



รูปที่ 2.16 GY-302 BH1750FVI เซนเซอร์วัดค่าแสง

ที่มา : (<https://www.arduinoall.com/article/306/70-arduino-เซ็นเซอร์วัดค่าแสง>)

ตารางที่ 2.4 แสดงชื่อขาสัญญาณของบอร์ด

Pin No.	ชื่อขาสัญญาณ	คำอธิบาย
1	GND	ขากาวด์
2	ADD	ขาสำหรับเลือก Address ของบอร์ด LOW=0x23 , HIGH =0x5C
3	SDA	ขาสัญญาณ SDA ของการเชื่อมต่อแบบ I2C
4	SCL	ขาสัญญาณ SCL ของการเชื่อมต่อแบบ I2C
5	VCC	ขาไฟเลี้ยงช่วงแรงดันที่ใช้ 3V-5V

2.6 Servo Motor

เซอร์โวมอเตอร์ (Servo Motor) เป็นมอเตอร์ที่มีการควบคุมการเคลื่อนที่ของมัน (State) ไม่ว่าจะเป็นระยะ ความเร็ว มุมการหมุน โดยใช้การควบคุมแบบป้อนกลับ (Feedback control) เป็นอุปกรณ์ที่สามารถควบคุมเครื่องจักรกล หรือระบบการทำงานนั้นๆ ให้เป็นไปตามความต้องการ เช่น ควบคุมความเร็ว (Speed), ควบคุมแรงบิด (Torque), ควบคุมแรงตำแหน่ง (Position), ระยะทางในการเคลื่อนที่ (มุม) (Position Control) ของตัวมอเตอร์ได้ ซึ่งมอเตอร์ทั่วไปไม่สามารถควบคุมในลักษณะงานเบื้องต้นได้ โดยให้ผลลัพธ์ตามความต้องการที่มีความแม่นยำสูง



รูปที่ 2.17 เซอร์โวมอเตอร์ (Servo Motor)

ที่มา : ([http://www.advance-electronic.com/blog/detail/86/th/-\(Servo-Motor\).html](http://www.advance-electronic.com/blog/detail/86/th/-(Servo-Motor).html))

2.6.1.1 ประเภทของเซอร์โวมอเตอร์

โดยทั่วไปจะมีทั้งดีซีและเอซีเซอร์โว ในเครื่องจักรรุ่นเก่าๆเราจะพบว่า DC Servo Motor มีการใช้เครื่องจักรกลอุตสาหกรรมมากกว่า AC Servo Motor เนื่องจากช่วงที่ผ่านมารควบคุมกระแสกระแสสูงๆ นั้นจะต้องใช้ SCRs แต่ปัจจุบันทรานซิสเตอร์ได้พัฒนาขีดความสามารถให้ตัดต่อกระแสสูงและใช้งานที่ความถี่ได้สูงขึ้น จึงทำให้ระบบควบคุมทางเอซีและระบบเซอร์โวได้ถูกนำมาใช้งานมากขึ้น ซึ่งสามารถแยกประเภทของเซอร์โวได้ดังนี้

2.6.1.1 มอเตอร์ชนิดที่มีแปรงถ่าน - เซอร์โวมอเตอร์ชนิดนี้ที่สเตเตอร์จะเป็นแม่เหล็กถาวร ส่วนโรเตอร์ยังใช้แปรงถ่านและคอมมิวเตอรียงกระแสเข้าสู่คลวคอาร์เมเจอร์ เหมือนกับดีซีมอเตอร์ทั่วไป

2.6.1.2 เซอร์โวมอเตอร์ชนิดที่ไม่มีแปรงถ่าน - เซอร์โวมอเตอร์ในกลุ่มนี้ประกอบด้วยดีซีเซอร์โว (DC Brushless Servo โรเตอร์ทำด้วยแม่เหล็กถาวร) เอซีเซอร์โว (AC Servo) ซึ่งมีทั้งแบบซิงโครนัสเซอร์โว อะซิงโครนัสเซอร์โว (การนำอินดักชันมอเตอร์มาใช้ทำเป็นระบบขับเคลื่อนเซอร์โวมอเตอร์) และ สเตปป์ิงเซอร์โวมอเตอร์

2.6.2 โครงสร้างของเซอร์โวมอเตอร์



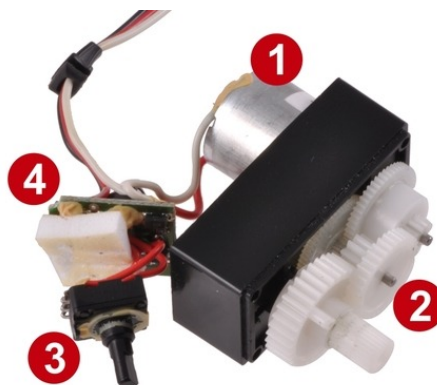
รูปที่ 2.18 ส่วนประกอบของเซอร์โวมอเตอร์

ที่มา : (<http://www.Arduino.cc/en/reference/servo>)

- Case ตัวถัง หรือ กรอบของตัว Servo Motor
- Mounting Tab ส่วนจับยึดตัว Servo กับชิ้นงาน
- Output Shaft เพลาส่งกำลัง
- Servo Horns ส่วนเชื่อมต่อกับ Output shaft เพื่อสร้างกลไก
- Cable สายเชื่อมต่อเพื่อ จ่ายไฟฟ้า และ ควบคุม Servo Motor จะประกอบด้วยสายไฟ 3 เส้น และใน RC Servo Motor จะมีสีของสายแตกต่างกันไปดังนี้

- สายสีแดง คือ ไฟเลี้ยง (4.8-6V)
- สายสีดำ หรือ น้ำตาล คือ กราวด์
- สายสีเหลือง (ส้ม ขาว หรือฟ้า) คือ สายส่งสัญญาณพัลส์ควบคุม (3-5V)
- Connector จุดเชื่อมต่อสายไฟ

2.6.3 ส่วนประกอบภายใน RC Servo Motor



รูปที่ 2.19 ส่วนประกอบภายใน RC Servo Motor

ที่มา : (<http://www.pololu.com/>)

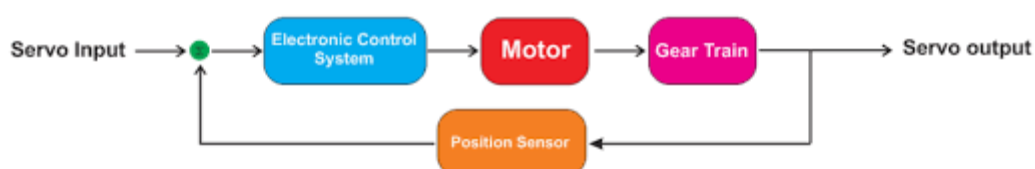
หมายเลข 1 คือ ดีซีมอเตอร์

หมายเลข 2 คือ ชุดเฟืองเกียร์ทดแรงมอเตอร์

หมายเลข 3 คือ ชุดควบคุมและปรับแต่งให้ค่าเอาต์พุตมีค่าเท่ากับ หรือใกล้เคียงกับค่าอินพุต

หมายเลข 4 คือ ส่วนที่ควบคุมและประมวลผล

2.6.4 บล็อกไดอะแกรมของ Servo Motor



รูปที่ 2.20 บล็อกไดอะแกรมของ Servo Motor

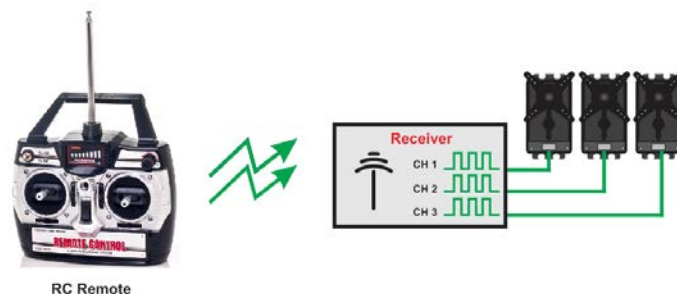
ที่มา : (www.Ebldc.com/?p=48)

- Motor เป็นส่วนของตัวมอเตอร์
- Gear Train หรือ Gearbox เป็นชุดเกียร์ทดแรง
- Position Sensor เป็นเซ็นเซอร์ตรวจจับตำแหน่งเพื่อหาค่าองศาในการหมุน
- Electronic Control System เป็นส่วนที่ควบคุมและประมวลผล

2.6.5 หลักการทำงานของ RC Servo Motor

เมื่อจ่ายสัญญาณพัลส์เข้ามายัง RC Servo Motor ส่วนวงจรควบคุม (Electronic Control System) ภายใน Servo จะทำการอ่านและประมวลผลค่าความกว้างของสัญญาณพัลส์ที่ส่งเข้ามาเพื่อแปลค่าเป็นตำแหน่งองศาที่ต้องการให้ Motor หมุนเคลื่อนที่ไปยังตำแหน่งนั้น แล้วส่งคำสั่งไปทำการควบคุมให้ Motor หมุนไปยังตำแหน่งที่ต้องการ โดยมี Position Sensor เป็นตัวเซ็นเซอร์คอยวัดค่ามุมที่ Motor กำลังหมุนเป็น Feedback กลับมาให้วงจรควบคุมเปรียบเทียบกับค่าอินพุตเพื่อควบคุมให้ได้ตำแหน่งที่ต้องการอย่างถูกต้องแม่นยำ สัญญาณ RC ในรูปแบบ PWM

ตัว RC Servo Motor ออกแบบมาใช้สำหรับรับคำสั่งจาก Remote Control ที่ใช้ควบคุมของเล่นด้วยสัญญาณวิทยุต่างๆ เช่นเครื่องบินบังคับ รถบังคับ เรือบังคับ เป็นต้น ซึ่ง Remote จำพวกนี้ที่ภาครับจะแปลงความถี่วิทยุออกมาในรูปแบบสัญญาณ PWM (Pulse Width Modulation)

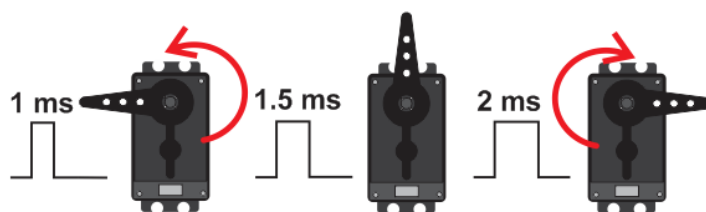


รูปที่ 2.21 Remote Control RC Servo Motor

ที่มา : (www.Learn.parallax.com/KickStart/900-00008)

มุมหรือองศาจะขึ้นอยู่กับความกว้างของสัญญาณพัลส์ ซึ่งโดยส่วนมากความกว้างของพัลส์ที่ใช้ใน RC Servo Motor จะอยู่ในช่วง 1-2 ms หรือ 0.5-2.5 ms

ตัวอย่างเช่นหากกำหนดความกว้างของสัญญาณพัลส์ไว้ที่ 1 ms ตัว Servo Motor จะหมุนไปทางซ้ายสุด ในทางกลับกันหากกำหนดความกว้างของสัญญาณพัลส์ไว้ที่ 2 ms ตัว Servo Motor จะหมุนไปยังตำแหน่งขวาสุด แต่หากกำหนดความกว้างของสัญญาณพัลส์ไว้ที่ 1.5 ms ตัว Servo Motor ก็จะหมุนมาอยู่ที่ตำแหน่งตรงกลางพอดี

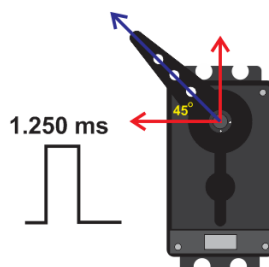


รูปที่ 2.22 มุมหรือองศาหมุนขึ้นอยู่กับความกว้างของสัญญาณพัลส์

ที่มา : (www.Learn.parallax.com/KickStart/900-00008)

ดังนั้นสามารถกำหนดองศาการหมุนของ RC Servo Motor ได้โดยการเทียบค่า เช่น RC Servo Motor สามารถหมุนได้ 180 องศา โดยที่ 0 องศาใช้ความกว้างพัลส์เท่ากับ 1000 us ที่ 180 องศาความกว้างพัลส์เท่ากับ 2000 us เพราะฉะนั้นค่าที่เปลี่ยนไป 1 องศาจะใช้ความกว้างพัลส์ต่างกัน $(2000-1000)/180$ เท่ากับ 5.55 us

จากการหาความกว้างพัลส์ที่มุม 1 องศาข้างต้น หากต้องกำหนดให้ RC Servo Motor หมุนไปที่มุม 45 องศาจะหาพัลส์ที่ต้องการได้จาก 5.55×45 เท่ากับ 249.75 us แต่ที่มุม 0 องศาเราเริ่มที่ความกว้างพัลส์ 1ms หรือ 1000 us เพราะฉะนั้นความกว้างพัลส์ที่ใช้กำหนดให้ RC Servo Motor หมุนไปที่ 45 องศา คือ $1000 + 249.75$ เท่ากับประมาณ 1250 us

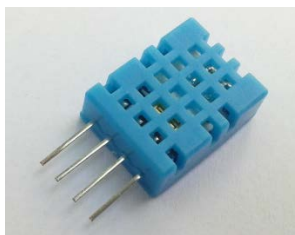


รูปที่ 2.23 RC Servo Motor หมุนไปที่ 45 องศา

ที่มา : (www.Learn.parallax.com/KickStart/900-00008)

2.7 Temperature Sensor

Temperature Sensor คือ โมดูลหรือเซ็นเซอร์สำหรับวัดอุณหภูมิและความชื้นในอากาศ ที่มีราคาถูก ใช้งานง่ายและสามารถใช้งานกับ Arduino Uno R3 ได้ ซึ่งจะมีอยู่สองแบบ คือแบบที่มาเป็นโมดูลกับแบบที่มีแค่เซ็นเซอร์มาให้อย่างเดียว โดยการรับส่งข้อมูลนั้นจะใช้สายสัญญาณเส้นเดียวกันและเป็นสัญญาณแบบดิจิตอล



รูปที่ 2.24 DHT-11 วัดอุณหภูมิและความชื้น

ที่มา : (<https://www.ioxhop.com/product/57/dht-11-%E0%B8%A7%E0%B8%B1>)

คุณสมบัติ

- ใช้แรงดันไฟฟ้า 3 ถึง 5V
- ใช้กระแสไฟฟ้าสูงสุด 2.5mA (ขณะทำการวัดค่า)
- เหมาะสำหรับวัดความชื้นระดับ 20-80% โดยมีความผิดพลาดในการวัดไม่เกิน 5%
- เหมาะสำหรับวัดอุณหภูมิ 0-50°C โดยมีความผิดพลาดในการวัดไม่เกิน $\pm 2^{\circ}\text{C}$
- ความถี่ในการวัด 1 Hz (อ่านค่าได้วินาทีละครั้ง)
- ขนาด 15.5mm x 12mm x 5.5mm
- 4 pins ใช้พื้นที่ในการวางขา 0.1"

วิธีการส่งข้อมูลของ DHT อุปกรณ์ตัวนี้ใช้การสื่อสารกับ MCU ของเราด้วยวิธี Single-wire Two-way Serial interface คือ การสื่อสารอนุกรมสองทางโดยใช้สายเส้นเดียว หรือ การสื่อสารใช้สายสื่อสารเพียงเส้นเดียวและส่งข้อมูลได้ทั้งจาก MCU ไปที่ตัว DHT11 และในทางกลับกันก็ได้

Arduino เปรียบเสมือน Master และ DHT11 เปรียบเสมือน Slave เกี่ยวกับ Pull up resistor การใช้ pull up resistor นั้นหมายถึงเราต้องการให้เวลาทำงานปกติในขณะที่ไม่มีอุปกรณ์ที่ต่ออยู่กับสายสื่อสารของเรา ระดับแรงดันของสัญญาณในสาย Data คือแรงดันระดับ "สูง" และจะมีแรงดันในระดับต่ำเมื่อมีอุปกรณ์ (จะเป็น Master หรือ Slave ก็ได้) ดึงสัญญาณลงในระดับ "ต่ำ" ดังนั้นหากเรามี Scope วัดระดับสัญญาณได้เป็น "สูง" ตลอดเวลา ก็หมายความว่าอุปกรณ์ของเราอาจจะผิดปกติ

ในการสื่อสารโดยใช้สายเส้นเดียวนั้น จำเป็นต้องใช้โปรโตคอลที่ตกลงกันไว้ระหว่างตัว MCU และ อุปกรณ์ที่ต้องการสื่อสารด้วย (Slave) อันดับแรก Master หรือ Arduino ของเราจะส่ง Start signal ที่เป็นแรงดันไฟฟ้าระดับต่ำอย่างน้อย 18 ไมโครวินาที ไปที่ Slave เพื่อให้ Slave เข้าใจว่าจะเริ่มส่งแล้ว แล้วรอไป 20-40 ไมโครวินาทีเพื่อรอ Slave ตอบกลับและเพื่อให้ Master รู้ว่า Slave ก็พร้อม Slave จะส่งแรงดันระดับต่ำกลับไปบ้าง การส่งแรงดันจาก Slave กลับไปจะนาน 80 ไมโครวินาที จากนั้นจะรออีก 80 ไมโครวินาที ก่อนที่จะส่งข้อมูลบิตแรก

2.8 หลักการเขียนโปรแกรม Arduino

ภาษาซีของ Arduino จะจัดรูปแบบโครงสร้างของการเขียนโปรแกรมออกเป็นส่วนย่อย ๆ หลายๆ ส่วน โดยเรียกแต่ละส่วนว่า ฟังก์ชัน และเมื่อนำฟังก์ชันมารวมเข้าด้วยกัน ก็จะเรียกว่า “โปรแกรม” โดยโครงสร้างการเขียนโปรแกรมของ Arduino นั้น ทุก ๆ โปรแกรมจะต้องประกอบไปด้วยฟังก์ชันจำนวนเท่าใดก็ได้ แต่อย่างน้อยที่สุดต้องมีฟังก์ชัน จำนวน 2 ฟังก์ชัน คือ setup() และ loop()

2.8.1 โครงสร้างการเขียนโปรแกรม Arduino

```
#include<servo.h>           //เรียกไลบรารี ชื่อ servo.h เข้ามาใช้ในโปรแกรม
int Servo1=9;               //กำหนดให้ Servo1 แทน Pin Digital-9
Servo myservo;              //สร้าง object ชื่อ myservo เพื่อควบคุม Servo
void setup(){myservo.attach(Serv1); } //กำหนดให้ใช้ขา Digital-9 สร้างสัญญาณ
ควบคุม Servo
void loop(){myservo.write(180); } //กำหนดค่าตำแหน่งให้กับ Servo = 180 องศา
```

จะได้โครงสร้างพื้นฐานของภาษาซีที่ใช้กับ Arduino นั้นจะประกอบไปด้วย 3 ส่วนใหญ่ๆ ด้วยกัน คือ

2.8.1.1 Header ในส่วนนี้จะมีหรือไม่มีก็ได้ ถ้ามีต้องกำหนดไว้ในส่วนเริ่มต้นของโปรแกรม ซึ่งส่วนของ Header ได้แก่ ส่วนที่เป็น Compiler Directive ต่าง ๆ รวมไปถึงส่วนของการประกาศตัวแปร และค่าคงที่ต่าง ๆ ที่จะใช้ในโปรแกรม

2.8.1.2 setup() ในส่วนนี้เป็นฟังก์ชันบังคับที่ต้องกำหนดให้มีในทุก ๆ โปรแกรม ถึงแม้ว่าในบางโปรแกรมจะไม่ต้องการใช้งานก็ยังจำเป็นต้องประกาศไว้ด้วยเสมอ เพียงแต่ไม่ต้องเขียนคำสั่งใด ๆ ไว้ในระหว่างวงเล็บปีกกา {} ที่ใช้เป็นตัวกำหนดขอบเขตของฟังก์ชัน โดยฟังก์ชันนี้จะใช้สำหรับบรรจุคำสั่งในส่วนที่ต้องทำให้โปรแกรมทำงานเพียงรอบเดียวตอนเริ่มต้นทำงานของโปรแกรมครั้งแรกเท่านั้น ซึ่งได้แก่คำสั่งเกี่ยวกับการ Setup ค่าการทำงานต่าง ๆ เช่น การกำหนดหน้าที่การใช้งานของ PinMode และการกำหนดค่า Baudrate สำหรับใช้งานพอร์ตสื่อสารอนุกรม เป็นต้น

2.8.1.3 loop() เป็นส่วนฟังก์ชันบังคับที่ต้องกำหนดให้มีในทุก ๆ โปรแกรม เช่นเดียวกันกับฟังก์ชัน setup() โดยฟังก์ชัน loop() นี้จะใช้บรรจุคำสั่งที่ต้องการให้โปรแกรมทำงานซ้ำ ๆ กันไปไม่รู้จบ ซึ่งถ้าเปรียบเทียบกับรูปแบบของ ANSI-C ส่วนนี้ก็คือ ฟังก์ชัน main() นั่นเอง

บทที่ 3

วิธีการดำเนินโครงการ

ในการดำเนินการโครงการเรื่องระบบควบคุมอุณหภูมิภายในโรงเรียนอจรรย์ยะมีขั้นตอนการสร้างในส่วนต่าง ๆ โดยทางกลุ่มผู้สร้างได้ร่วมกันวางแผนในการปฏิบัติงานและจัดการบางงานตามความเหมาะสม ขั้นตอนในการดำเนินโครงการ แบ่งออกเป็นดังนี้

3.1 การวางแผนและการเตรียมการ

3.2 การออกแบบชิ้นงาน

3.3 การดำเนินการสร้างระบบควบคุมอุณหภูมิภายในโรงเรียนอจรรย์ยะ

3.1 การวางแผนและการเตรียมการ

การวางแผนและการเตรียมการเริ่มเมื่อคณะกรรมการพิจารณาโครงการให้เสนอหัวข้อโครงการในภาคเรียนที่ 1 ทางกลุ่มผู้จัดทำได้เสนอหัวข้อโครงการระบบควบคุมอุณหภูมิภายในโรงเรียนอจรรย์ยะ ซึ่งมีลำดับขั้นตอนต่าง ๆ ในการดำเนินโครงการดังตารางที่ 3.1

3.1.1 การวางแผนทำโครงการ

3.1.1.1 เสนอหัวข้อโครงการกับอาจารย์ที่ปรึกษา

3.1.1.2 หาข้อมูลเสนอเพิ่มเติม เรื่องการพัฒนาชุดคำสั่ง

3.1.1.3 อนุมัติโครงการโดยอาจารย์ที่ปรึกษา

3.1.1.4 เขียนโครงการบทที่ 1-3

3.1.1.5 ศึกษาและหาข้อมูล โดยศึกษาจากหนังสือวิจัย เว็บไซต์ต่าง ๆ เป็นต้น

3.1.1.6 ออกแบบโรงเรียน

3.1.1.7 วางแผนการปฏิบัติงาน โดยจัดลำดับก่อนและหลังการปฏิบัติงาน

3.1.1.8 ส่งโครงการบทที่ 1- 2 ให้อาจารย์ที่ปรึกษา

3.1.1.9 ซ้อมชิ้นส่วนประกอบต่าง ๆ

3.1.1.10 ทำการประกอบชิ้นส่วนต่าง ๆ

3.1.1.11 ทำการศึกษาอุปกรณ์ที่สร้าง

3.1.1.12 ทำการสร้างโค้ดเขียนโปรแกรม

3.1.1.13 ทำการใส่โค้ดโปรแกรมลงในอุปกรณ์

3.1.1.14 ตบแต่งชิ้นส่วนต่าง ๆ ในอุปกรณ์จนครบทุกส่วน

3.1.1.15 ทดลองการใช้อุปกรณ์จริง

3.1.1.16 เขียนโครงการบทที่ 3-5 โดยนำข้อมูลที่ได้จากการสร้างชิ้นงาน,การออกแบบ,ผลการทดลองใช้งาน, การนำปัญหาที่ประสบในการดำเนินโครงการและข้อเสนอแนะต่าง ๆ มาเรียงให้ได้ใจความสมบูรณ์

3.1.1.17 ส่งโครงการบทที่ 1-5 ให้อาจารย์ที่ปรึกษาตรวจสอบความถูกต้อง

3.1.1.18 ยื่นขอสอบโครงการ หลังจากทฤษฎี บทที่1-5 ได้ผ่านการตรวจสอบความถูกต้อง

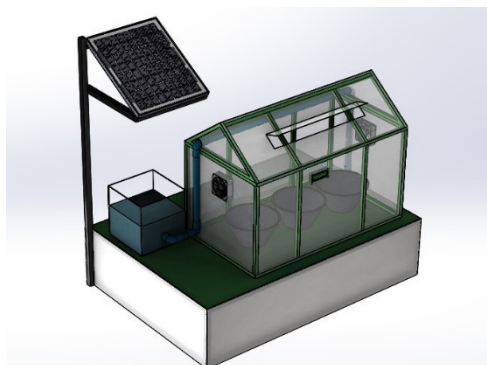
3.1.1.19 อนุมัติสอบโครงการ

3.1.1.20

3.1.1.21 ส่งโครงการโดยนำเนื้อหาบทที่ 1-5 เข้าเล่ม แล้วนำไปให้คณะกรรมการสอบโครงการไว้เป็นตัวอย่างในการศึกษาและหลักฐาน

3.2 การออกแบบชิ้นงาน

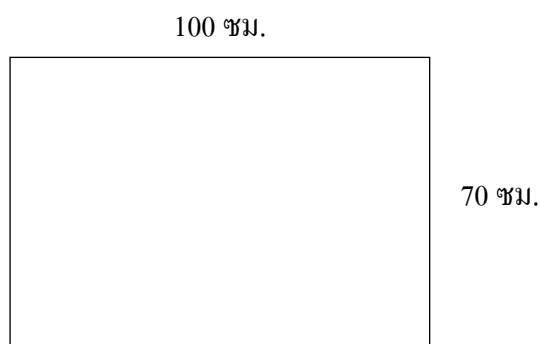
3.2.1 ออกแบบภายนอกโรงเรือน



รูปที่ 3.1 ออกแบบภายนอกโรงเรือน

3.2.2 ออกแบบฐานโรงเรือน

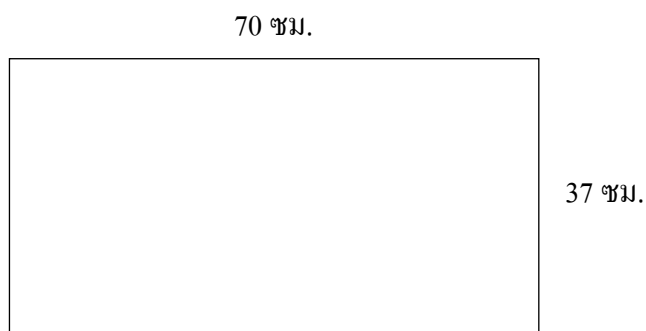
โครงสร้างฐานโรงเรือนมีขนาดความกว้าง 70 เซนติเมตรและความยาว 100 เซนติเมตร



รูปที่ 3.2 โครงสร้างฐานโรงเรือน

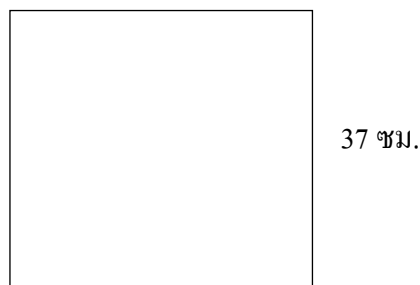
3.2.3 ออกแบบโครงสร้างโรงเรือน

3.2.3.1 ชั้นส่วนด้านหน้าและหลังโรงเรือนมีขนาดความกว้าง 37 เซนติเมตรและความยาว 70 เซนติเมตร



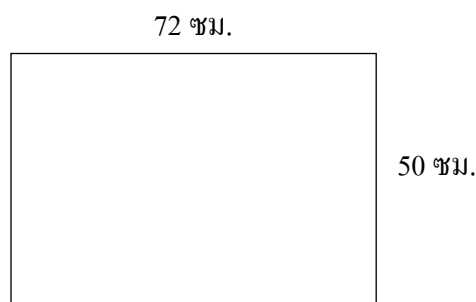
รูปที่ 3.3 โครงสร้างโรงเรือนชั้นส่วนด้านหน้าและหลัง

3.2.3.2 ชั้นส่วนด้านข้างโรงเรียนมีขนาดความกว้าง 37 เซนติเมตรและความยาว 40 เซนติเมตร



รูปที่ 3.4 โครงสร้างโรงเรียนชั้นส่วนด้านข้าง

3.2.3.3 ชั้นส่วนหลังคาโรงเรียนมีขนาดความกว้าง 50 เซนติเมตรและความยาว 72 เซนติเมตร



รูปที่ 3.5 โครงสร้างโรงเรียนชั้นส่วนหลังคาโรงเรียน

3.3 การดำเนินการสร้างระบบรดน้ำต้นไม้อัตโนมัติและแอปพลิเคชันควบคุมโรงเรียนอัจฉริยะ

3.3.1 ขั้นตอนการสร้างโรงเรียน

การสร้างโครงสร้างโรงเรียน เพื่อที่จะให้ได้ฐานหรือโครงสร้าง ที่มีความแข็งแรงสวยงาม และมีความกระชับเมื่อติดตั้งอุปกรณ์เข้าไปเพิ่มเติม ซึ่งประกอบไปกล่องควบคุมหรือคอนโทรลระบบทุกอย่างที่ติดไว้ด้านล่างขวาของฐานโรงเรียน เพื่อให้สามารถกันน้ำที่กระเด็นมาจากการรดน้ำต้นไม้ได้ มีจอ LCD แสดงผลที่ด้านหน้าโรงเรียน มีสวิทช์ ปิด-เปิด อยู่ด้านข้างของฐาน มีระบบระบายความร้อนด้วยพัดลมทั้ง 2 ด้านของโรงเรียน และหลังคาเปิดปิดได้อยู่ด้านบน อีกทั้งยังสร้างให้ยึดติดกับโครงเหล็กอย่างมาตรฐาน สามารถถอดประกอบได้



รูปที่ 3.6 ขั้นตอนการสร้างโรงเรือน

3.3.2 ขั้นตอนการประกอบและติดตั้ง

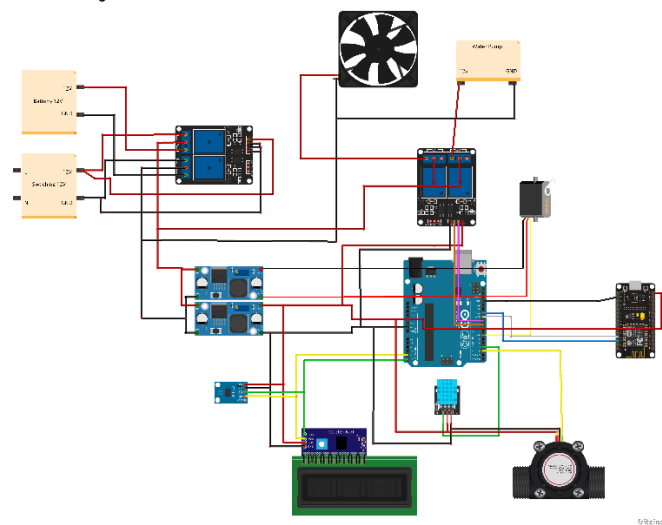
ในการประกอบและติดตั้ง เราจะใช้ arduino ในการสั่งงานเพื่อควบคุมอุปกรณ์ 4 อย่างคือ เซ็นเซอร์วัดความอุณหภูมิในอากาศ เซ็นเซอร์วัดค่าแสง ระบบเซอร์โวมอเตอร์ และระบบพัดลมระบายความร้อน หลักการทำงานของระบบระบายความร้อน เริ่มจากระบบของมอเตอร์เซอร์โว มาช่วยในเรื่องการระบายความร้อนโดยการเปิดและปิดหลังคาโรงเรือน โดยหากค่าแสงเกินกว่าที่กำหนดให้หลังคาทำการเปิดเองอัตโนมัติและหากค่าแสงต่ำกว่าที่กำหนดให้หลังคาทำการปิดเองอัตโนมัติ , ระบบระบายความร้อนด้วยพัดลม โดยใช้เซ็นเซอร์วัดอุณหภูมิมาวัดอุณหภูมิ หากอุณหภูมิสูงกว่าที่กำหนดไว้ ให้พัดลมทำงานเองอัตโนมัติ และหากอุณหภูมิต่ำกว่าที่กำหนดไว้ ให้พัดลมปิดการทำงานเองอัตโนมัติ แล้วนำข้อมูลมาแสดงผลผ่านจอ LCD ช่วยให้สามารถรับรู้ค่าอุณหภูมิและค่าแสงของสภาพแวดล้อมต่าง ๆ ภายในระบบควบคุมโรงเรือนเพาะปลูก



รูปที่ 3.7 ขั้นตอนการประกอบและติดตั้ง

3.3.3 ขั้นตอนการต่อวงจร

การต่อวงจรภายในระบบดังรูป 3.9



รูปที่ 3.8 ขั้นตอนการต่อวงจร

3.3.4 ขั้นตอนการเขียนโปรแกรมคำสั่ง

```

Sketch, Feb 19 - ESP Resistor | Arduino 1.8.10
ไฟล์ sketch ใหม่: แก้ไขใหม่ หรือลบทิ้ง

#include <ESP8266WiFi.h>
#include <SoftwareSerial.h>
#include <WiFiClient.h>
#include <BlynkSimpleEsp8266.h>

#define BLYNK_PRINT Serial

SoftwareSerial NodeSerial(D0, D3); // RX | TX

char token [] = "4c31K1ZD9qkonQisE1RfN0#Q6pyjWOWC"; //w5sueqj0X8M2q7hb0c2d_N2XN6FedyV8c
char ssid [] = "wifi5baht";
char pass [] = "12345678";

BlynkTimer SmartFarm;
void setup() {
  pinMode(D0, OUTPUT);
  pinMode(D3, OUTPUT);
  //Blynk.begin(token,ssid,pass);
  Blynk.begin(token,ssid,pass,"blynk.honey.co.th",8080);

  // Blynk.begin(token,ssid,pass,"blynkfree.info",8080);
  Serial.begin(9600);

  // Use the serial() for sketch

```

รูปที่ 3.9 ขั้นตอนการเขียนโปรแกรมคำสั่ง

บทที่ 4

ผลการศึกษา

ผลการศึกษาทำให้เราได้รู้ถึงการทำงานร่วมกันเป็นกลุ่มต้องสามัคคีกัน ในการวิเคราะห์ และออกแบบตลอดจนถึงขั้นตอนการทำงานแต่ละขั้นตอนว่ามีอะไรบ้าง ในที่นี้ทางคณะผู้จัดทำจะกล่าวถึงผลของการศึกษาข้อมูล และผลที่ได้รับอย่างละเอียด แบ่งออกเป็นดังนี้

4.1 การทดสอบวัดอุณหภูมิและระบบระบายความร้อนเองอัตโนมัติด้วยพัดลม

4.2 การทดสอบวัดค่าแสงและระบบระบายความร้อนเองอัตโนมัติด้วยมอเตอร์เซอร์โว

4.1 การทดสอบวัดอุณหภูมิและระบบระบายความร้อนเองอัตโนมัติด้วยพัดลม

4.1.1 อุปกรณ์การทดลอง

4.1.1.1 ไมโครคอนโทรลเลอร์ Node MCU

4.1.1.2 เครื่องคอมพิวเตอร์ที่ติดตั้งโปรแกรม Arduino

4.1.1.3 เซนเซอร์วัดอุณหภูมิ

4.1.1.4 พัดลมระบายความร้อน

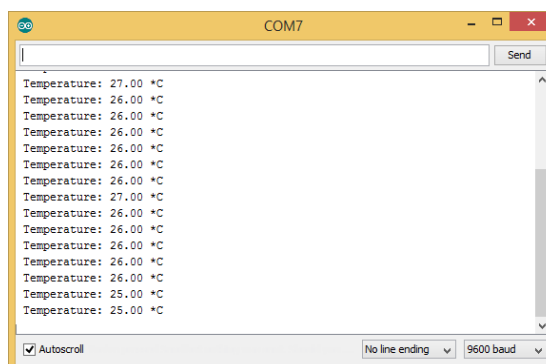
4.1.1.5 LCD Display

4.1.2 วิธีการทดลอง

4.1.2.1 เปิดโปรแกรม Arduino IDE ที่เขียนชุดคำสั่งไว้ จากนั้นเรียกใช้ Serial Monitor โดยให้คลิกที่ Tools > Serial Monitor หรือกดปุ่ม Ctrl+Shift+M

4.1.2.2 อ่านค่าที่ได้รับจากเซนเซอร์วัดความชื้นในอากาศจาก Serial Monitor, จอ LCD และดูการเปิด-ปิดเองอัตโนมัติของพัดลมระบายความร้อน

4.1.2.3 นำมาเปรียบเทียบกับอุปกรณ์วัดอุณหภูมิ



รูปที่ 4.1 อ่านค่าอุณหภูมิในอากาศจาก Serial Monitor



รูปที่ 4.2 อ่านค่าอุณหภูมิในอากาศจาก LCD



รูปที่ 4.3 ภาพแสดงการทำงานของพัดลมระบายความร้อน

4.1.3 ผลการทดลอง

ตารางที่ 4.1 ผลการทดลองการวัดอุณหภูมิขณะอากาศปกติ

อ่านค่าอุณหภูมิในอากาศ จาก Serial Monitor (%)	อ่าน ค่า อุณหภูมิ ใน อากาศจากจอ LCD (%)	ค่าความคลาด เคลื่อน (%)	การทำงานของพัด ลมระบายความร้อน
25.20	25.00	0.79	ไม่ทำงาน
25.19	25.00	0.75	ไม่ทำงาน
25.10	25.00	0.40	ไม่ทำงาน
26.35	26.00	1.33	ไม่ทำงาน
25.95	25.00	3.66	ไม่ทำงาน
24.85	25.00	0.60	ไม่ทำงาน
26.00	26.00	0.00	ไม่ทำงาน
27.10	27.00	0.37	ไม่ทำงาน
26.42	26.00	1.59	ไม่ทำงาน
25.11	25.00	0.44	ไม่ทำงาน

การทดลองการวัดอุณหภูมิขณะอากาศปกติ ค่าความคลาดเคลื่อนเฉลี่ย = 0.87%

และระบบระบายความร้อนเองอัตโนมัติด้วยพัดลมไม่ทำงานเลย

ตารางที่ 4.2 ผลการทดลองการวัดอุณหภูมิขณะอากาศร้อน

อ่านค่าอุณหภูมิในอากาศ จาก Serial Monitor (%)	อ่าน ค่า อุณหภูมิ ใน อากาศจากจอ LCD (%)	ค่าความคลาด เคลื่อน (%)	การทำงานของพัด ลมระบายความร้อน
38.20	39.00	2.09	ทำงาน
39.05	39.00	0.13	ทำงาน
40.10	40.00	0.25	ทำงาน
41.85	41.00	2.03	ทำงาน
42.00	43.00	2.38	ทำงาน
43.91	44.00	0.20	ทำงาน
44.50	45.00	1.12	ทำงาน
44.60	45.00	0.90	ทำงาน
43.01	43.00	0.02	ทำงาน
44.08	45.00	2.09	ทำงาน

การทดลองการวัดอุณหภูมิขณะอากาศร้อน ค่าความคลาดเคลื่อนเฉลี่ย = 0.64%

และระบบระบายความร้อนเองอัตโนมัติด้วยพัดลมทำงานตลอดเวลาที่อุณหภูมิร้อนเกินกว่า 40 %

4.2 การทดสอบวัดค่าแสงและระบบระบายความร้อนเองอัตโนมัติด้วยมอเตอร์เซอร์โว

4.2.1 อุปกรณ์การทดลอง

4.2.1.1 ไมโครคอนโทรลเลอร์ Node MCU

4.2.1.2 เครื่องคอมพิวเตอร์ที่ติดตั้งโปรแกรม Arduino

4.2.1.3 เซนเซอร์วัดค่าแสง

4.2.1.4 มอเตอร์เซอร์โว

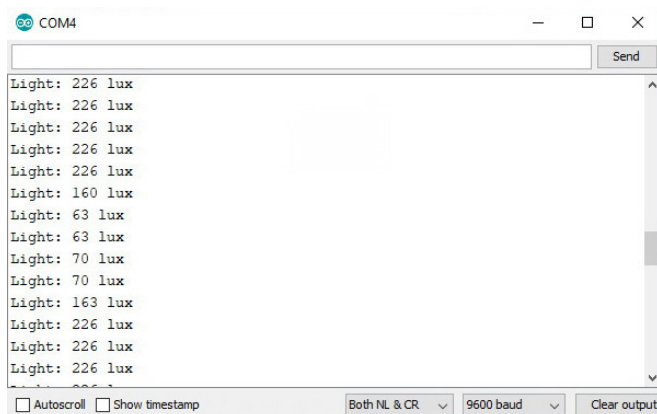
4.2.1.5 LCD Display

4.2.2 วิธีการทดลอง

4.2.2.1 เปิดโปรแกรม Arduino IDE ที่เขียนชุดคำสั่งไว้ จากนั้นเรียกใช้ Serial Monitor โดยให้คลิกที่ Tools > Serial Monitor หรือกดปุ่ม Ctrl+Shift+M

4.2.2.2 อ่านค่าที่ได้รับจากเซนเซอร์วัดค่าแสงจาก Serial Monitor, จอ LCD และดูการเปิด-ปิดเองอัตโนมัติของหลังคาโรงเรือน

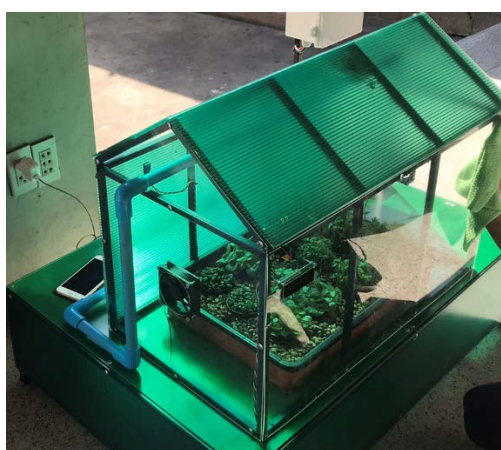
4.1.2.3 นำมาเปรียบเทียบกับอุปกรณ์วัดค่าแสง



รูปที่ 4.4 อ่านค่าแสงจาก Serial Monitor



รูปที่ 4.5 อ่านค่าแสงจาก LCD



รูปที่ 4.6 ภาพแสดงการทำงานของหลังคาโรงเรือน

4.2.3 ผลการทดลอง

ตารางที่ 4.3 ผลการทดลองการวัดค่าแสงช่วงเวลา 8.00 – 12.00 น.

เวลาวัด ค่าแสง	อ่านค่าแสงจาก Serial Monitor (%)	อ่านค่าแสงจากจอ LCD (%)	ค่าความคลาด เคลื่อน (%)	การทำงานของหลังคา โรงเรือน
9.00	587	579	1.36	ไม่ทำงาน
9.10	602	645	1.83	ไม่ทำงาน
9.20	612	613	0.16	ไม่ทำงาน
9.30	700	723	3.29	ไม่ทำงาน
10.00	513	523	1.95	ไม่ทำงาน
10.10	698	702	0.57	ไม่ทำงาน
10.20	753	752	0.13	ไม่ทำงาน
10.30	736	745	1.22	ไม่ทำงาน
11.00	864	858	0.69	ทำงาน
11.10	980	981	0.10	ทำงาน

การทดลองการวัดค่าแสงช่วงเวลา 8.00 – 12.00 น. ค่าความคลาดเคลื่อนเฉลี่ย = 0.69%

และระบบระบายความร้อนด้วยหลังคาโรงเรือนเปิดขึ้น เมื่อค่าแสงเกิน 700 Lux

ตารางที่ 4.4 ผลการทดลองการวัดค่าแสงช่วงเวลา 12.01 - 15.00 น.

เวลาวัด ค่าแสง	อ่านค่าแสงจาก Serial Monitor (%)	อ่านค่าแสงจากจอ LCD (%)	ค่าความคลาด เคลื่อน (%)	การทำงานของหลังคา โรงเรือน
12.01	980	985	0.51	ทำงาน
12.10	1020	1011	0.88	ทำงาน
12.30	1052	1052	0.00	ทำงาน
13.00	1100	1090	0.91	ทำงาน
13.10	1035	1030	0.48	ทำงาน
13.20	756	759	0.40	ทำงาน
13.30	893	893	0.00	ทำงาน
14.00	840	845	0.60	ทำงาน
14.30	790	794	0.51	ทำงาน
15.00	750	751	0.13	ทำงาน

การทดลองการวัดค่าแสงช่วงเวลา 12.01 – 15.00 น. ค่าความคลาดเคลื่อนเฉลี่ย = 0.01%
และระบบระบายความร้อนด้วยหลังคาโรงเรือนเปิดขึ้นตลอด

บทที่ 5

สรุป อภิปรายผล และข้อเสนอแนะ

ในการดำเนินโครงการระบบควบคุมอุณหภูมิภายในโรงเรียนอจรรย์ชะ นั้นสามารถบรรลุวัตถุประสงค์ที่วางไว้ ซึ่งโปรแกรมสามารถทำงานได้อย่างที่ต้องการ แต่การดำเนินการโครงการก็ประสบปัญหาต่าง ๆ หลายอย่าง ซึ่งผู้ดำเนินการโครงการมีข้อเสนอแนะที่จะนำมาใช้พัฒนาปรับปรุงแก้ไขให้ระบบควบคุมอุณหภูมิภายในโรงเรียนอจรรย์ชะ สามารถทำงานเองอัตโนมัติโดยอิงจากค่าแสงและอุณหภูมิ และแสดงข้อมูลผ่าน LCD เพื่อให้มีประสิทธิภาพการทำงานของโปรแกรมดีที่สุด

5.1 วัตถุประสงค์ของโครงการ

- 5.1.1 เพื่อออกแบบและสร้างระบบควบคุมอุณหภูมิภายในโรงเรียนอจรรย์ชะ
- 5.1.2 เพื่อพัฒนาผลงานที่ประดิษฐ์ขึ้นให้เข้าสู่ความเป็นมาตรฐานสามารถนำไปใช้งานได้ อย่างมีคุณภาพประหยัดและปลอดภัย
- 5.1.3 เพื่อให้ผู้จัดทำโครงการนำความรู้จากการศึกษาในหลักสูตรของประกาศนียบัตรวิชาชีพชั้นสูงสาขาวิชาเทคโนโลยีสารสนเทศ มาใช้ในการประกอบอาชีพและการพัฒนาวิชาชีพของตน

5.2 ประโยชน์ที่ได้รับจากการศึกษา

- 5.2.1 ระบบระบายความร้อนโรงเรียนเพาะปลูกสามารถทำงานได้อย่างอัตโนมัติ
- 5.2.2 ระบบสามารถแสดงค่าแสงและค่าอุณหภูมิภายในโรงเรียน
- 5.2.3 ผู้จัดทำโครงการนำความรู้จากการศึกษาไปประยุกต์ใช้ในชีวิตประจำวัน

5.3 ปัญหาที่ประสบในการดำเนินโครงการ

ในการดำเนินการโครงการระบบรดน้ำต้นไม้อัตโนมัติและแอปพลิเคชันควบคุมโรงเรียนอจรรย์ชะ นั้นทางคณะผู้จัดทำโครงการจะอธิบายสาเหตุ และวิธีการแก้ปัญหาเป็นข้อ ๆ ดังนี้

- 5.3.1 ปัญหาในการจัดซื้ออุปกรณ์ต่าง ๆ และตัวบอร์ด
- 5.3.2 ปัญหาด้านการศึกษาแผงวงจร
- 5.3.3 ปัญหาด้านการศึกษาชุดคำสั่งควบคุม
- 5.3.4 ปัญหาด้านการเชื่อมต่ออุปกรณ์
- 5.3.5 ปัญหาด้านงบประมาณการจัดทำ

5.3.6 ปัญหาด้านมอเตอร์เซอร์โวไม่สามารถยกหลังคาโรงเรือนได้อย่างมีประสิทธิภาพ

5.4 ผลการดำเนินโครงการ

ในการดำเนินการโครงการระบบควบคุมอุณหภูมิภายในโรงเรือนอัจฉริยะ เริ่มจากการนำเสนอโครงการต่อคณะกรรมการพิจารณาทางคณะกรรมการได้เสนอแนะส่วนต่าง ๆ และคณะผู้จัดทำได้ศึกษาข้อมูลรายละเอียดต่าง ๆ เกี่ยวกับการทำโครงการระบบควบคุมอุณหภูมิภายในโรงเรือนอัจฉริยะ โดยได้ทำการออกแบบและดำเนินการจัดทำตามที่วางไว้จนสำเร็จ

ผลการดำเนินการโครงการระบบควบคุมอุณหภูมิภายในโรงเรือนอัจฉริยะ ตัวแผงวงจรนี้สามารถส่งสัญญาณและเชื่อมต่อเข้าไปในแผงวงจรได้โดยการเชื่อมต่อสัญญาณ WIFI เข้ากับแผงควบคุมได้สำเร็จ

5.5 อภิปรายผล

จากผลการดำเนินการโครงการระบบควบคุมอุณหภูมิภายในโรงเรือนอัจฉริยะ ถือว่าประสบความสำเร็จตามที่ตั้งจุดประสงค์ไว้ คือระบบระบายความร้อนโรงเรือนเพาะปลูกสามารถทำงานได้อย่างอัตโนมัติ และแสดงค่าแสงและค่าอุณหภูมิภายในโรงเรือนได้ นอกจากนี้คณะผู้จัดทำยังได้รับความรู้และประสบการณ์ในการทำโครงการนี้เป็นอย่างมาก

5.6 ข้อเสนอแนะ

ในที่นี้จะกล่าวถึงข้อเสนอแนะ ในการพัฒนาและต่อยอดของระบบควบคุมอุณหภูมิภายในโรงเรือนอัจฉริยะ

5.6.1 ข้อเสนอแนะทั่วไป

5.6.1.1 ควรออกแบบให้โรงเรือนมีขนาดเล็กกว่านี้

5.6.1.2 ควรออกแบบโรงเรือนให้ตรงกับทฤษฎีทางสถาปัตยกรรม

5.6.2 ข้อเสนอแนะทางเทคนิค

5.6.2.1 ควรติดตั้งระบบพัดลมให้ระบายความร้อนไปในทิศทางเดียวกัน

5.6.2.2 ควรติดตั้งเซอร์โว 2 ตัว เพื่อให้สามารถยกหลังคาได้อย่างมีประสิทธิภาพ

บรรณานุกรม

- กมลชนก ลอมโสม. (2560). การควบคุมหุ่นยนต์ผ่านเครือข่ายไร้สายโดยใช้อาร์ดูโอโนปริยญา
นิพนธ์. นครนายก: ภาควิชาวิศวกรรมไฟฟ้า คณะวิศวกรรมศาสตร์ มหาวิทยาลัยศรี
นครินทร์วิโรฒองครักษ์.
- กอบเกียรติ สระอุบล. (2561). พัฒนา IoT บนแพลตฟอร์ม Arduino และ Raspberry Pi.
กรุงเทพมหานคร:อินเตอร์มีเดีย.
- ดอนสัน ปงผาบ. (2561). ภาษาซีและ Arduino (พิมพ์ครั้งที่ 1). กรุงเทพมหานคร: สำนักพิมพ์
สมาคมส่งเสริมเทคโนโลยี (ไทย-ญี่ปุ่น).
- เดชฤทธิ์ มณีธรรม. (2560). คัมภีร์และการใช้งานไมโครคอนโทรลเลอร์ Arduino. กรุงเทพมหานคร:
ซีเอ็ดยูเคชั่น.
- ประภาส พุ่มพวง. (2561). การใช้งาน ESP32 เบื้องต้น. กรุงเทพมหานคร: ซีเอ็ดยูเคชั่น.

ภาคผนวก

ภาคผนวก ก	แบบเสนอร่างโครงการ
ภาคผนวก ข	รายงานผลความก้าวหน้าโครงการ
ภาคผนวก ค	ประวัติผู้เขียน

ภาคผนวก ก

แบบเสนอร่างโครงการ

ภาคผนวก ข

รายงานผลความก้าวหน้าโครงการ

ภาคผนวก ค

ประวัติผู้เขียน



สาขาวิชาเทคโนโลยีสารสนเทศ

แบบเสนอร่างโครงการ

เรื่อง

ระบบควบคุมอุณหภูมิภายในโรงเรือนอัจฉริยะ

Temperature Control System in Intelligent House

โดย

นายกฤษฎา ขาวบู่ รหัสประจำตัว 41269

นายณรงค์ มุลศิริ รหัสประจำตัว 40978

ภาคเรียนที่ 1/2562

สาขาวิชาเทคโนโลยีสารสนเทศ

วิทยาลัยเทคโนโลยีอรรถวิทย์พัฒนวิชาการ

แบบเสนอร่างโครงการ
สาขาวิชาเทคโนโลยีสารสนเทศ

ชื่อโครงการ ระบบควบคุมอุณหภูมิภายในโรงเรือนอัจฉริยะ
Temperature Control System in Intelligent House

ชื่อผู้เสนอโครงการ 1. นายกฤษฎา ขาวมู รหัสประจำตัว 41269 (หัวหน้ากลุ่มโครงการ)

ชื่อผู้ร่วมโครงการ 2. นายณรงค์ มุลศิริ รหัสประจำตัว 40978

นักศึกษาสาขาวิชาเทคโนโลยีสารสนเทศ รอบวันอาทิตย์*

ชื่ออาจารย์ที่ปรึกษาโครงการ อาจารย์คุณานนท์ สุขเกษม

มีความประสงค์ขออนุมัติหัวข้อโครงการ เพื่อเป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
ประกาศนียบัตรวิชาชีพชั้นสูง ในวิชาโครงการ จำนวน 4 หน่วยกิต

จึงเรียนมาเพื่อโปรดพิจารณา ดังรายละเอียดโครงการที่แนบมาด้วย

ลงชื่อ (หัวหน้ากลุ่มโครงการ)

ลงชื่อ (สมาชิกกลุ่มโครงการ)

..... / /

ความเห็นอาจารย์ที่ปรึกษาโครงการ	ความเห็นผู้รับผิดชอบโครงการ สาขาวิชา
.....	
.....	
.....	
ลงนาม	ลงนาม
..... / /	 / /
ลงนาม	
..... / /	

1. ชื่อโครงการ ระบบควบคุมอุณหภูมิภายในโรงเรือนอัจฉริยะ

Temperature Control System in Intelligent House

2. ความเป็นมาและความสำคัญของปัญหา

ปัจจุบันสภาพอากาศค่อนข้างมีความแปรปรวนและยากที่จะควบคุม ส่งผลกระทบทั้งต่อการดำเนินชีวิตของมนุษย์ สัตว์ สิ่งแวดล้อม รวมถึงพืช แต่ข้อจำกัดของพืชคือไม่สามารถเลือกสภาพแวดล้อมที่เหมาะสมในการเจริญเติบโตได้อย่างมนุษย์หรือสัตว์ ดังนั้นเพื่อควบคุมปัจจัยด้านสภาพแวดล้อมในการเพาะปลูกพืชให้มีความเหมาะสมนั้นจึงมีการพัฒนาการปลูกพืชด้วยการปลูกในระบบโรงเรือน แต่ปัญหาการปลูกพืชในโรงเรือนในประเทศไทยคือ “อุณหภูมิหรือความร้อนสะสมภายในโรงเรือน” โดยเฉพาะโรงเรือนที่ไม่มี การระบายความร้อน มีโอกาสที่อุณหภูมิภายในจะสูงถึง 50 องศาเซลเซียสซึ่งมีผลต่อการเจริญเติบโตของพืชอย่างมาก ดังนั้นการเลือกโรงเรือนจึงมีความสำคัญเนื่องจากโรงเรือนแต่ละแบบมีความสามารถในการระบายอากาศและการลดอุณหภูมิได้แตกต่างกัน

รศ.ดร.ธรรมศักดิ์ ทองเกตุ อาจารย์ภาควิชาพืชสวน คณะเกษตร มหาวิทยาลัยเกษตรศาสตร์ กล่าวว่า ประเทศไทยตั้งอยู่ในเขตร้อน การปลูกพืชจึงมีโอกาที่จะเกิดโรคและแมลงศัตรูรบกวนได้ตลอด โดยเฉพาะการปลูกพืชนอกโรงเรือนที่ไม่สามารถควบคุมสภาพแวดล้อมต่าง ๆ ได้ การผลิตพืชในระบบปิด หรือการปลูกพืชในโรงเรือน จึงเป็นคำตอบที่จะช่วยให้เกษตรกรสามารถผลิตพืชได้ทั้งปริมาณ คุณภาพ และปลอดภัยได้อย่างต่อเนื่องตลอดทั้งปี ภายใต้การควบคุมปัจจัยแวดล้อมที่เหมาะสม เช่น ฝน พายุ ความแปรปรวนของสภาพอากาศ ป้องกันการระบาดของโรคและแมลงศัตรูลดการใช้สารเคมี นอกจากนี้การปลูกพืชในโรงเรือนยังช่วยเพิ่มประสิทธิภาพในการให้น้ำทางน้ำ (Fertigation System) เป็นระบบการผลิตพืชที่เป็นมิตรกับสิ่งแวดล้อม

ผู้วิจัยมีแนวคิดที่จะนำเอาเทคโนโลยีไอโอที (IOT) มาประดิษฐ์ระบบควบคุมอุณหภูมิภายในโรงเรือนอัจฉริยะ โดยนำระบบของมอเตอร์เซอร์โว มาช่วยในเรื่องการระบายความร้อนโดยการเปิดและปิดหลังคาโรงเรือน โดยหากค่าแสงเกินกว่าที่กำหนดให้หลังคาทำการเปิดเองอัตโนมัติ และหากค่าแสงต่ำกว่าที่กำหนดให้หลังคาทำการปิดเองอัตโนมัติ , ระบบระบายความร้อนด้วยพัดลม โดยใช้เซนเซอร์วัดอุณหภูมิมาวัดอุณหภูมิ หากอุณหภูมิสูงกว่าที่กำหนดไว้ ให้พัดลมทำงานเองอัตโนมัติ และหากอุณหภูมิต่ำกว่าที่กำหนดไว้ ให้พัดลมปิดการทำงานเองอัตโนมัติ แล้วนำข้อมูลมาแสดงผลผ่านจอ LCD ช่วยให้ผู้สามารถรับรู้ค่าอุณหภูมิและค่าแสงของสภาพแวดล้อมต่าง ๆ ภายในระบบควบคุมโรงเรือนเพาะปลูกเพื่อเป็นการลดสภาพความแปรปรวนของดินฟ้าอากาศและเพื่อเพิ่มผลผลิตที่มากขึ้น โดยที่ความร้อนภายในระบบควบคุมโรงเรือนเพาะปลูกจะอยู่ในช่วงที่เหมาะสมเพื่อให้การดูแลระบบสมาร์ทฟาร์มจำลองเป็นไปอย่างอัตโนมัติและมีประสิทธิภาพ

3. วัตถุประสงค์ของโครงการ

- 3.1) เพื่อออกแบบและสร้างระบบควบคุมอุณหภูมิภายในโรงเรือนอัจฉริยะ
- 3.2) เพื่อพัฒนาผลงานที่ประดิษฐ์ขึ้นให้เข้าสู่ความเป็นมาตรฐานสามารถนำไปใช้งานได้
อย่างมีคุณภาพประหยัดและปลอดภัย
- 3.3) เพื่อให้ผู้จัดทำโครงการนำความรู้จากการศึกษาในหลักสูตรของประกาศนียบัตรวิชาชีพชั้นสูงสาขาวิชาเทคโนโลยีสารสนเทศ มาใช้ในการประกอบอาชีพและการพัฒนา
วิชาชีพของตน

4. ขอบเขตของโครงการ

- 4.1) สามารถปิด-เปิดการทำงานของหลังคาโรงเรือนได้อัตโนมัติ
- 4.2) สามารถปิด-เปิดการทำงานของพัดลมระบายความร้อนโรงเรือนได้อัตโนมัติ
- 4.3) สามารถแสดงค่าแสงและค่าอุณหภูมิภายในโรงเรือน

5. ขั้นตอนการดำเนินงาน

- 5.1) คิดหัวข้อโครงการ
- 5.2) จัดทำเอกสารแบบเสนอร่างโครงการ
- 5.3) ศึกษาการทำงานของมอเตอร์เซอร์โว
- 5.4) ปรับปรุงแก้ไข
- 5.5) ศึกษาเอกสารที่เกี่ยวข้อง
- 5.6) วิเคราะห์และออกแบบระบบ
- 5.7) สร้างอุปกรณ์ฮาร์ดแวร์
- 5.8) ทดสอบระบบ
- 5.9) ตรวจสอบความเรียบร้อยของระบบ
- 5.10) ส่งให้ที่ปรึกษาตรวจสอบ
- 5.11) นำเสนอโครงการ
- 5.12) จัดทำรูปเล่มโครงการ

7. ประโยชน์ที่คาดว่าจะได้รับ

- 7.1) ระบบระบายความร้อนโรงเรือนเพาะปลูกสามารถทำงานได้อย่างอัตโนมัติ
- 7.2) ระบบสามารถแสดงค่าแสงและค่าอุณหภูมิภายในโรงเรือน
- 7.3) ผู้จัดทำโครงการนำความรู้จากการศึกษาไปประยุกต์ใช้ในชีวิตประจำวัน

8. งบประมาณที่ใช้ในการทำโครงการ

8.1) ค่าพิมพ์เอกสาร	200 บาท
8.2) ค่าทำเล่มเอกสาร โครงการ	200 บาท
8.3) ค่ากระดาษ	200 บาท
8.4) ค่าอุปกรณ์ในการทำโครงงาน	2,000 บาท
รวมค่าใช้จ่ายทั้งหมด	2,600 บาท

เอกสารอ้างอิง

<https://www.arduinoall.net/arduino-tutor/>

<http://www.dpu.ac.th/bigdata/iot-smart-agriculture.html>

<https://www.baanlaesuan.com/59796/design/design-update/places/deva-farm>



แบบประเมินความก้าวหน้าโครงการ

ระบบควบคุมอุณหภูมิภายในโรงเรียนอัจฉริยะ
Temperature Control System in Intelligent House

ชื่อผู้จัดทำ

นายกฤษฎา ขาวบู่ รหัสประจำตัว 41269
นายณรงค์ มุลศิริ รหัสประจำตัว 40978

ภาคเรียนที่ 1/2562

สาขาวิชาเทคโนโลยีสารสนเทศ
วิทยาลัยเทคโนโลยีอรรณพวิทยพัฒน์

แบบฟอร์มประเมินความก้าวหน้าโครงการ

ชื่อโครงการ ระบบควบคุมอุณหภูมิภายในโรงเรียนอัสสัมชัญ
ชื่อโครงการ Temperature Control System in Intelligent House

ปีการศึกษา 2562

ชื่อผู้จัดทำโครงการ (1) นายกฤษฎา ขาวบู่ รหัส 41269 ชั้นปี ปวส.2/35
(2) นายณรงค์ มูลศิริ รหัส 40978 ชั้นปี ปวส.2/35

ชื่ออาจารย์ที่ปรึกษาโครงการ อาจารย์คุณานนท์ สุขเกษม

โครงการนี้จัดอยู่ในกลุ่มของ

- | | |
|--|--|
| ☐ Web Programming | ☐ Computer Multimedia |
| ☒ Computer Programming | ☐ Database System |
| ☐ Hardware Computer | ☐ |

ขอบเขตของโครงการทั้งหมด

ระบบงานสามารถตรวจสอบผ่านทางโปรแกรมอาคูนและจอ LCD ได้ มีการกำหนดขอบเขตของระบบควบคุมอุณหภูมิภายในโรงเรียนอัสสัมชัญ ดังต่อไปนี้

1. สามารถปิด-เปิดการทำงานของหลังคาโรงเรียนได้อัตโนมัติ
2. สามารถปิด-เปิดการทำงานของพัดลมระบายความร้อนโรงเรียนได้อัตโนมัติ
3. สามารถแสดงค่าแสงและค่าอุณหภูมิภายในโรงเรียน

รายงานผลความก้าวหน้าของโครงการ 25%

รายละเอียดของหัวข้อที่รายงานผลความก้าวหน้า	ผ่าน	ไม่ผ่าน	เหตุผล (กรณีไม่ผ่าน)
1. รวบรวมข้อมูล			
2. ศึกษาข้อมูล - Farm System - การทำงานมอเตอร์เซอร์โว - อุปกรณ์ฮาร์ดแวร์ที่ใช้			
3. เขียนโปรแกรม			
4. โครงการ บทที่ 1			
5. โครงการ บทที่ 2 (บางส่วน)			

การประเมินผลความก้าวหน้าของโครงการโดยอาจารย์ที่ปรึกษา

☐ ผ่าน

☐ ไม่ผ่าน

ข้อชี้แจงของอาจารย์ที่ปรึกษา (กรณีที่อาจารย์ที่ปรึกษาประเมินไม่ผ่าน)

.....
.....

ลายเซ็น
()

อาจารย์ที่ปรึกษา

วันที่ / / 25.....

ส่วนนี้สำหรับกรรมการ

การประเมินผลความก้าวหน้าของโครงการโดยกรรมการ

☐ ผ่าน

☐ ไม่ผ่าน

ความคิดเห็นของกรรมการ (กรณีที่กรรมการประเมินไม่ผ่าน)

.....
.....

ลายเซ็น
()

กรรมการตรวจสอบ

วันที่ / / 25.....

รายงานผลความก้าวหน้าของโครงการ 50%

รายละเอียดของหัวข้อที่รายงานผลความก้าวหน้า	ผ่าน	ไม่ผ่าน	เหตุผล (กรณีไม่ผ่าน)
ส่วนโครงสร้างโรงเรียนอจรรย์ยะ			
6. สร้างฐานโรงเรียน			
7. สร้างหลังคาโรงเรียน			
8. ประกอบและติดตั้ง			
ส่วนของจอ LCD			
9. แสดงค่าแสง			
10. แสดงอุณหภูมิภายในโรงเรียน			
11. จัดทำเอกสารโครงการบทที่ 2 ในส่วนที่เหลือ			
12. จัดทำเอกสารโครงการบทที่ 3			

การประเมินผลความก้าวหน้าของโครงการโดยอาจารย์ที่ปรึกษา

☐ ผ่าน

☐ ไม่ผ่าน

ข้อชี้แจงของอาจารย์ที่ปรึกษา (กรณีที่อาจารย์ที่ปรึกษาประเมินไม่ผ่าน)

.....

ลายเซ็น

()

อาจารย์ที่ปรึกษา

วันที่ / / 25.....

ส่วนนี้สำหรับกรรมการ

การประเมินผลความก้าวหน้าของโครงการโดยกรรมการ

☐ ผ่าน

☐ ไม่ผ่าน

ความคิดเห็นของกรรมการ (กรณีที่กรรมการประเมินไม่ผ่าน)

.....
.....

ลายเซ็น

()

กรรมการตรวจสอบ

วันที่ / / 25.....

รายงานผลความก้าวหน้าของโครงการ 75%

รายละเอียดของหัวข้อที่รายงานผลความก้าวหน้า	ผ่าน	ไม่ผ่าน	เหตุผล (กรณีไม่ผ่าน)
13. ต่อบรรณภายในระบบ			
14. เชื่อมต่อกับระบบโรงเรียน			
15. ทดสอบการใช้ระบบงาน			
16. แก้ไขระบบงาน			
17. จัดทำโครงการบทที่ 4			

การประเมินผลความก้าวหน้าของโครงการโดยอาจารย์ที่ปรึกษา

☐ ผ่าน

☐ ไม่ผ่าน

ข้อชี้แจงของอาจารย์ที่ปรึกษา (กรณีที่อาจารย์ที่ปรึกษาประเมินไม่ผ่าน)

.....

ลายเซ็น

()

อาจารย์ที่ปรึกษา

วันที่ / / 25.....

ส่วนนี้สำหรับกรรมการ

การประเมินผลความก้าวหน้าของโครงการโดยกรรมการ

☐ ผ่าน

☐ ไม่ผ่าน

ความคิดเห็นของกรรมการ (กรณีที่กรรมการประเมินไม่ผ่าน)

.....

ลายเซ็น

()

กรรมการตรวจสอบ

วันที่ / / 25.....

รายงานผลความก้าวหน้าของโครงการ 100%

รายละเอียดของหัวข้อที่รายงานผลความก้าวหน้า	หมายเหตุ
18. ทดสอบระบบการทำงาน	
19. แก้ไขและสรุบบางงาน	
20. รูปเล่มโครงการฉบับสมบูรณ์	
21. ขอสอบโครงการ	

การประเมินผลความก้าวหน้าของโครงการโดยอาจารย์ที่ปรึกษา

☐ ผ่าน

☐ ไม่ผ่าน

ข้อชี้แจงของอาจารย์ที่ปรึกษา (กรณีที่อาจารย์ที่ปรึกษาประเมินไม่ผ่าน)

.....

ลายเซ็น

()

อาจารย์ที่ปรึกษา

วันที่ / / 25.....

บันทึกการเข้าพบอาจารย์ที่ปรึกษา

[illegible]



ประวัติผู้เขียน

ชื่อ-สกุล	นาย กฤษฎา ขาวบู่
วันเดือนปีเกิด	14 ตุลาคม 2535
สถานที่เกิด	บุรีรัมย์
สถานที่อยู่ปัจจุบัน	123 หมู่ 8 อาคาร โชคชัยบุญทวี แมนชั่น ต.สำโรงกลาง อ.พระประแดง จ.สมุทรปราการ 10130
สถานศึกษา	หลักสูตรประกาศนียบัตรวิชาชีพชั้นสูง สาขาวิชาเทคโนโลยีสารสนเทศ วิทยาลัยเทคโนโลยีอรรถวิทย์พัฒนวิชาการ
ประวัติการศึกษา	
พ.ศ.2553	มัธยมศึกษาตอนปลาย โรงเรียนหนองหงส์พิทยาคม
พ.ศ.2557-ปัจจุบัน	หลักสูตรประกาศนียบัตรวิชาชีพชั้นสูง สาขาวิชาเทคโนโลยีสารสนเทศ วิทยาลัยเทคโนโลยีอรรถวิทย์พัฒนวิชาการ



ประวัติผู้เขียน

ชื่อ-สกุล	นาย ณรงค์ มุลศิริ
วันเดือนปีเกิด	5 เมษายน 2538
สถานที่เกิด	แพร่
สถานที่อยู่ปัจจุบัน	แสงไทยอพาร์ทเมนต์ 42 ซ.สุภาพงษ์ แขวงหนองบอน เขต
ประเทศ	กรุงเทพ 10250
สถานศึกษา	หลักสูตรประกาศนียบัตรวิชาชีพชั้นสูง สาขาวิชาเทคโนโลยีสารสนเทศ วิทยาลัยเทคโนโลยีอรรถวิทย์พัฒนวิชาการ
ประวัติการศึกษา	
พ.ศ 2559	การศึกษานอกระบบ
พ.ศ 2561 - ปัจจุบัน	หลักสูตรประกาศนียบัตรวิชาชีพชั้นสูง สาขาวิชาเทคโนโลยีสารสนเทศ วิทยาลัยเทคโนโลยีอรรถวิทย์พัฒนวิชาการ