การควบคุมและแจ้งเตือนระบบระบายอากาศของตู้ชุมสายโทรศัพท์อัตโนมัติ

Temperature Monitor and Alarm System for MSAN

กษิเดช ฤทธิ์งาม

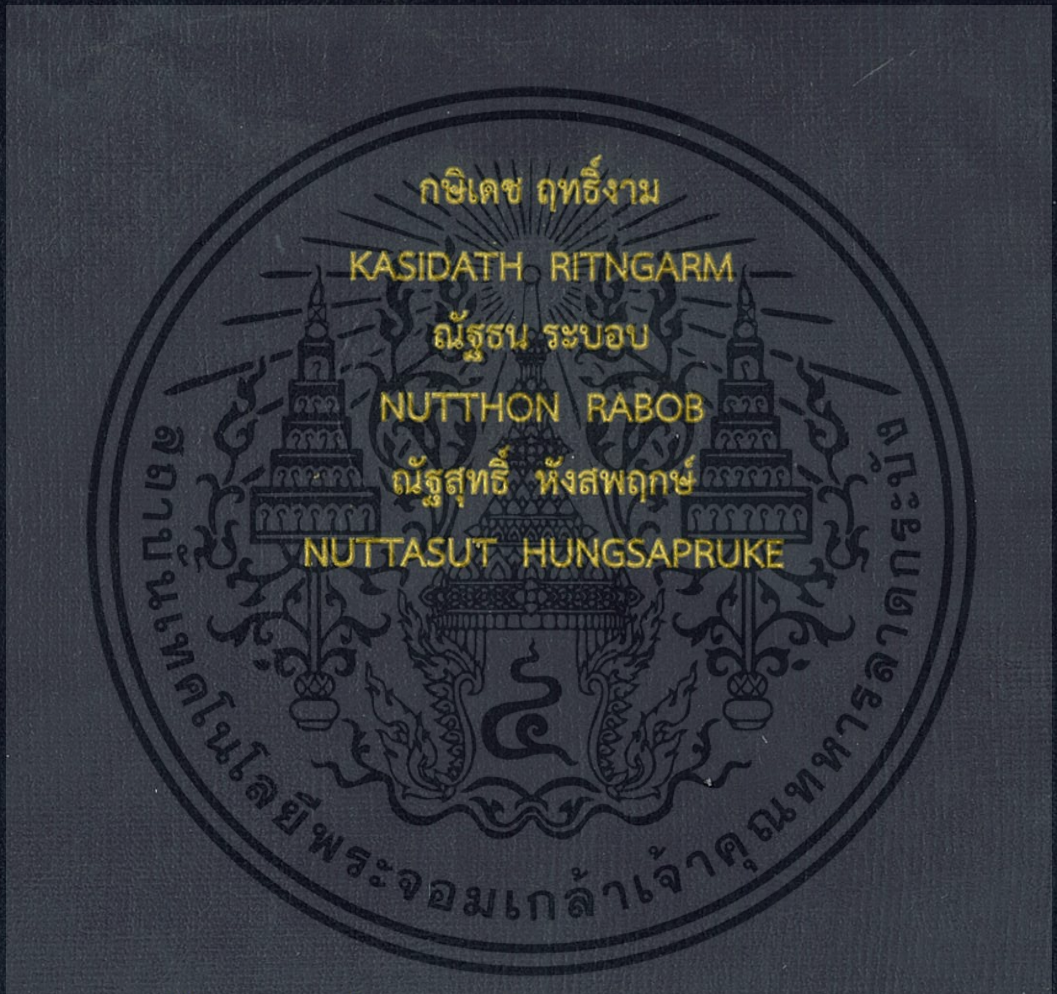
KASIDATH RITNGARM

ณัฐธน ระบอบ

NUTTHON RABOB

ณัฐสุทธิ์ หังสพฤกษ์

NUTTASUT HUNGSAPRUKE



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตร์บัณฑิต

สาขาวิชาวิศวกรรมอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2558

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

การควบคุมและแจ้งเตือนระบบระบายอากาศของตู้ชุมสายโทรศัพท์อัตโนมัติ

Temperature Monitor and Alarm System for MSAN

T143924

กษิเดช ฤทธิ์งาม
KASIDATH RITNGARM
ณัฐธน ระบอบ
NUTTHON RABOB
ณัฐสุทธิ์ หังสพฤกษ์
NUTTASUT HANGSAPRUKE

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เลขหมู่.....
เลขทะเบียน 143924
วัน.เดือน.ปี 04 ต.ค. 2559

b.12810757
i.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตร์บัณฑิต
สาขาวิชาวิศวกรรมอิเล็กทรอนิกส์
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2558

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การควบคุมและแจ้งเตือนระบบระบายอากาศของตู้ชุมสายโทรศัพท์อัตโนมัติ
Temperature Monitor and Alarm System for MSAN

โดย

กษิเดช ฤทธิ์งาม รหัส 55010059
ณัฐธน ระบอบ รหัส 55010360
ณัฐสุทธิ์ หังสพฤกษ์ รหัส 55010412

อาจารย์ที่ปรึกษา
ผศ.ดร. กิติพล ชิตสกุล

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตร์บัณฑิต
สาขาวิชาวิศวกรรมอิเล็กทรอนิกส์
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2558

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์	ปีการศึกษา 2558
ภาควิชา	วิศวกรรมอิเล็กทรอนิกส์
คณะ	วิศวกรรมศาสตร์
	สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
เรื่อง	การควบคุมและแจ้งเตือนระบบระบายอากาศของตู้ชุมสายโทรศัพท์อัตโนมัติ
	Temperature Monitor and Alarm System for MSAN
ผู้จัดทำ	นายกษิเดช ฤทธิ์งาม รหัส 55010059
	นายณัฐธน ระบอบ รหัส 55010360
	นายณัฐสุทธิ์ หังสพฤกษ์ รหัส 55010412

ปริญญานิพนธ์นี้ผ่านการตรวจโดยอาจารย์ที่ปรึกษาแล้ว

(ผศ.ดร. กิติพล ชิตสกุล)
อาจารย์ที่ปรึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อโครงงาน	การควบคุมและแจ้งเตือนระบบระบายอากาศของตู้ชุมสายโทรศัพท์อัตโนมัติ
นักศึกษา	นายกษิเดช ฤทธิ์งาม รหัสนักศึกษา 55010059
	นายณัฐธน ระบอบ รหัสนักศึกษา 55010360
	นายณัฐสุทธิ์ หังสพฤกษ์ รหัสนักศึกษา 55010412
ปริญญา	วิศวกรรมศาสตร์บัณฑิต
สาขาวิชา	วิศวกรรมอิเล็กทรอนิกส์
ปีการศึกษา	2558
อาจารย์ที่ปรึกษา	ผศ.ดร.กิติพล ชิตสกุล

บทคัดย่อ

โครงงานนี้ได้เป็นการพัฒนาและปรับปรุงระบบระบายอากาศในตู้ชุมสายโทรศัพท์ ซึ่ง ระบบเดิมนั้นได้เป็นการใช้พัดลมโดยความเร็วสูงสุดคงที่ ทำให้เกิดความเสียหายกับพัดลมเมื่อใช้เวลานาน ระบบการแจ้งเตือน ว่าเกิดปัญหาในระบบ จึงทำการปรับปรุงระบบระบายอากาศ โดยใช้ NodeMCU ในการควบคุมพัดลมในระบบระบายอากาศให้มีความเร็วตามอุณหภูมิภายในตู้ เพื่อเป็นการประหยัดพลังงาน และเพิ่มอายุการใช้งานของพัดลม อีกทั้ง ได้เพิ่มระบบแจ้งเตือน ใช้งานได้ผ่าน Website เพื่อบอกอุณหภูมิและความชื้น ภายในตู้และ ความเร็วของพัดลม ได้แบบ real-time และสามารถเก็บข้อมูลเป็นสถิติ เพื่อนำไปปรับปรุงพัฒนาระบบต่อได้อีกด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Project Title	Temperature Monitor and Alarm System for MSAN	
Student	Kasidath Ritngram	Student ID. 55010059
	Nutthon Rabob	Student ID. 55010360
	Nuttasut Hungsapruke	Student ID. 55010412

Degree	Bachelor of Engineering
Program	Electronics Engineering
Year	2015
Project Advisor	Asst.Prof.Dr.Kitiphol Chitsakul

ABSTRACT

This project improves the ventilator system in MSAN. For the old system works at the constant highest speed, ventilators are damaged through long time usage. NodeMCU is used to control ventilators so they work at the speed which corresponds with the temperature in MSAN. By this means, energy can be saved and ventilators can be used longer. Besides, this system works on Website telling temperature, humidity in MSAN and speed of ventilators in real time, which we can record them in statistic form for further system improvement.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

โครงงานชิ้นนี้สำเร็จได้ด้วยความช่วยเหลือจากอาจารย์ ผศ.ดร.กิติพล ชิตสกุล ซึ่งเป็นที่ปรึกษาในการทำโครงงานและช่วยหาวิธีแก้ปัญหาและอุปสรรคที่เกิดขึ้นขอบคุณอาจารย์ทุกท่านที่ช่วยให้ความรู้ และขอบคุณรุ่นพี่และเพื่อนๆที่ได้ให้ความช่วยเหลือและคำแนะนำเพิ่มเติมในการทำโครงงานชิ้นนี้สำเร็จลุล่วงได้ด้วยดี

กษิเดช ฤทธิ์งาม
ณัฐธน ระบอบ
ณัฐสุทธิ์ หังสพฤกษ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อ i
Abstract ii
กิตติกรรมประกาศ iii
สารบัญ ..... iv-vi
สารบัญรูปภาพ ..... vii-viii
สารบัญตาราง ix
บทที่ 1 บทนำ 1
1.1 ความเป็นมาของโครงงาน 1
1.2 วัตถุประสงค์ของโครงงาน 1
1.3 ขอบเขตของโครงงาน 1
1.4 ประโยชน์ที่ได้รับจากโครงงาน 1
1.5 วิธีการดำเนินการ 2
1.6 โครงสร้างของรายงาน 2
บทที่ 2 ทฤษฎีที่เกี่ยวข้อง 3
2.1 Microcontroller 3
2.2 Node MCU 5
2.3 เซนเซอร์ตรวจวัดอุณหภูมิ(Temperature Sensor) 7
2.4 มอเตอร์ไฟฟ้ากระแสตรง (DC Motor) 8
2.4.1 ส่วนที่อยู่กับที่หรือที่เรียกว่าสเตเตอร์ (Stator) 9
2.4.2. ตัวหมุน (Rotor) 9
2.5 อุปกรณ์ตู้ชุมสายโทรศัพท์และอินเตอร์เน็ต (MSAN Smart Node) 10
2.6 ทฤษฎีที่เกี่ยวข้อง 11
2.6.1 วงจรขยายสัญญาณโดยใช้ทรานซิสเตอร์แบบพื้นฐาน 11
2.6.1.1 วงจรขยายสัญญาณอิมิตเตอร์ร่วม 11
2.6.1.2 วงจรขยายแบบอิมิตเตอร์ร่วมที่มีการ Bypass 13
2.6.2 MOSFET 14
2.6.3 ไดโอด (Diode) 15
2.6.3.1คุณลักษณะเฉพาะของกระแสและแรงดัน 16
2.6.3.2 พฤติกรรมของสัญญาณขนาดเล็ก 17

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

หน้า

2.7 ภาษาที่ใช้เขียนเพื่อออกแบบโปรแกรม 17
2.7.1 ภาษาซี (C) 17
2.7.2 ไพทอน(Python) 18
2.7.2.1 ความสามารถของภาษา Python 18
2.7.2.2 หลักการทำงานของภาษา Python 19
2.7.3 ภาษา PHP 20
2.7.3.1 โครงสร้างของภาษา PHP 20
2.7.3.2 ความสามารถของภาษา PHP 21
2.8 โปรแกรมที่ใช้สำหรับการออกแบบ 22
2.8.1 Arduino (อาดูยโน่) 22
2.8.2 XAMPP 24
2.8.3 Netbeans 26
2.9 High Voltage step down convertor 27
บทที่ 3 การออกแบบ 28
3.1 การเก็บข้อมูลอุณหภูมิภายในและภายนอกของตู้ MSAN 28
3.1.1 การเก็บข้อมูลอุณหภูมิภายในของตู้MSAN 29
3.1.2 การเก็บข้อมูลอุณหภูมิภายนอกของตู้MSAN 29
3.2 โครสร้างการทำงานของระบบระบายอากาศ 30
3.3 การออกแบบส่วนควบคุมความเร็วพัดลมตามอุณหภูมิ 31
3.3.1 ส่วนวงจรควบคุมความเร็วพัดลมตามอุณหภูมิ 31
3.3.2 ผังงานโปรแกรมควบคุมความเร็วพัดลมตามอุณหภูมิ 32
3.4 การออกแบบส่วนวัดความเร็วรอบพัดลม 33
3.4.1 ผังงานโปรแกรมส่วนวัดความเร็วพัดลมตามอุณหภูมิ 35
3.5 การออกแบบส่วนแสดงข้อมูลบน Web Application 36

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

หน้า

บทที่ 4 ผลการทดลอง **37**
4.1 กราฟแสดงอุณหภูมิจากการจำลองตู้ MSAN 37
4.1.1 ค่าอุณหภูมิและเวลาที่นำไปพอทลงบนกราฟ 38
4.2 ผลการทดลองส่วนควบคุมพัดลมตามอุณหภูมิ 38
4.2.1 ที่อุณหภูมิ <40 c duty cycle ที่วัดได้เท่ากับ 20% 38
4.2.2 ที่อุณหภูมิ <45 c duty cycle ที่วัดได้เท่ากับ 50% 39
4.2.3 ที่อุณหภูมิ <50 c duty cycle ที่วัดได้เท่ากับ 80% 39
4.3 ผลการทดลองส่วนวัดความเร็วรอบพัดลม 40
4.4 ข้อมูลจาก database และค่าที่แสดงผลบน Web Application 40
4.5 ผลการส่งข้อมูลทาง E-mail 41
บทที่ 5 บทสรุป **42**
บรรณานุกรม **43**
ภาคผนวก **44**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ

รูปที่ **หน้า**

2.1Microcontroller....3
2.2 รูปบอร์ด Node MCU....5
2.3 รายละเอียดหน้าที่การทำงานของขาต่างๆของ Node MCU....6
2.4 Temperature Sensor (DHT11,DHT22)....7
2.5 Thermocouple....7
2.6 DC Fan motor....8
2.7 Stator....9
2.8 Rotor....9
2.9 ระบบเครือข่ายและอุปกรณ์ตู้ชุมสายโทรศัพท์และอินเตอร์เน็ต(MSAN)....10
2.10 ลักษณะของวงจรขยาย BJT 3 ชนิด....11
2.11 (รูปซ้าย) วงจรขยาย common emitter (รูปขวา) เป็นวงจรเทียบเท่า....12
2.12 (รูปซ้าย) วงจรขยาย bypassed common emitter (รูปขวา) เป็นวงจรเทียบเท่า....13
2.13 การทำงานของ Mosfet....14
2.14 diode และสัญญาลักษณ์ และ ขั้วของ Diode....16
2.15 พฤติกรรมของไดโอด....16
2.16 สัญญาลักษณ์ภาษา C....18
2.17 สัญญาลักษณ์ภาษา Python....19
2.18 ตัวอย่างการเขียนภาษา PHP....20
2.19 ตัวอย่างการเขียนภาษา PHP(ต่อ)....21
2.20 ตัวอย่างโปรแกรม Arduino....23
2.21 สัญญาลักษณ์โปรแกรม XAMPP....24
2.22 ตัวอย่างโปรแกรม XAMPP....25
2.23 ตัวอย่างโปรแกรม Netbeans....26
2.24 High voltage Step down convertor....27
3.1 ตู้ชุมสายที่ใช้ในการวัดอุณหภูมิ....28
3.2 อุณหภูมิภายในตู้ MSAN....29
3.3 อุณหภูมิภายนอกตู้ MSAN....29
3.4 โครงสร้างการทำงานของระบบระบายอากาศ....30
3.5 วงจรควบคุมความเร็วพัดลมตามอุณหภูมิ....31
3.6 ผังงานควบคุมความเร็วพัดลมตามอุณหภูมิ....32
3.7 ความสัมพันธ์ระหว่างความถี่และความเร็วรอบพัดลม....33
3.8 วงจรวัดความเร็วรอบพัดลมจากสัญญาณของพัดลม....34

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ(ต่อ)

รูปที่ **หน้า**

3.9 ผังงานการวัดความเร็วรอบของพัดลม....35
3.10 Block diagram sent data on Web Application....36
4.1 ความสัมพันธ์ระหว่างอุณหภูมิกับเวลาในตู้จำลอง....37
4.2 duty cycle ที่วัดได้เท่ากับ 20%....38
4.3 duty cycle ที่วัดได้เท่ากับ 50%....39
4.4 duty cycle ที่วัดได้เท่ากับ 80%....39
4.5 ข้อมูลจาก Database....40
4.6 ค่าที่แสดงผลทาง Web Application....41
4.7 ข้อมูลที่แจ้งเตือนมาทาง E-mail....41

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่ **หน้า**

3.1 ข้อมูล Volt ต่อ Frequency และค่า RPM.....33
4.1 ค่าอุณหภูมิและเวลาที่นำไปพอทลงบนความสัมพันธ์ระหว่างอุณหภูมิกับเวลา.....38
4.2 ผลการทดลองส่วนวัดความเร็วรอบพัดลม.....40

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความเป็นมาของโครงงาน

ในปัจจุบัน การสื่อสารเป็นสิ่งสำคัญมาก ไม่ว่าจะเป็นโทรศัพท์บ้าน หรืออินเตอร์เน็ต เป็นสิ่งที่เกือบทุกบ้านนั่นจะมี จากเหตุผลนี้ทำให้มีผู้ใช้บริการเป็นจำนวนมาก ทำให้เกิดชุมสายโทรศัพท์จำนวนมาก เพื่อให้การบริการครอบคลุมได้อย่างทั่วถึง ส่งผลทำให้การดูแลรักษาของตู้ชุมสายนั่นทำได้ลำบาก เนื่องจากพนักงานที่ดูแลรักษานั่นมีจำนวนน้อย ดังนั่นทางกลุ่มของผู้จัดทำ จึงเกิดความคิดในการแจ้งเตือน การชำรุดของระบบระบายอากาศของตู้ชุมสายโทรศัพท์ และพัฒนาการควบคุมพัดลมในระบบระบายอากาศเพื่อให้มีการประหยัดพลังงานอีกด้วย

1.2 วัตถุประสงค์ของโครงงาน

1.2.1 ออกแบบการควบคุมพัดลมในระบบระบายความร้อนในตู้ชุมสายโทรศัพท์

1.2.2 ออกแบบ และสร้างระบบแจ้งเตือนว่าพัดลมในตู้ชุมสายโทรศัพท์ชำรุด

1.2.3 นำข้อมูล อุณหภูมิ ความชื้นภายในตู้ชุมสายโทรศัพท์ และความเร็วรอบของพัดลมในระบบระบายความร้อน ส่งเข้าผ่านServerและแสดงผลข้อมูลผ่านทาง Web Application

1.3 ขอบเขตของโครงงาน

1.3.1 ควบคุมความเร็วของพัดลมในระบบระบายอากาศโดยขึ้นอยู่กับอุณหภูมิ

1.3.2 สามารถบอก อุณหภูมิ ความชื้นภายในตู้ชุมสายโทรศัพท์ และความเร็วรอบของพัดลมในระบบระบายความร้อนได้

1.3.3 นำข้อมูลจากข้อ 1.3.2 เข้าสู่ Web Application

1.4 ประโยชน์ที่ได้รับจากโครงงาน

1.4.1 ได้รับประสบการณ์เพิ่มเติม ในการออกแบบระบบ การเขียนชุดคำสั่ง และอุปกรณ์ในการใช้งาน

1.4.2 พบเห็นอุปสรรค ปัญหาจริง จากการออกแบบระบบ และนำความรู้จากห้องเรียนมาแก้ไขปัญหา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.5 วิธีการดำเนินการ

1.5.1 เก็บข้อมูลความร้อนในชุมสายโทรศัพท์ขณะทำงาน และสภาพแวดล้อมบริเวณตู้ชุมสายโทรศัพท์

1.5.2 ออกแบบระบบควบคุมพัดลมในระบบระบายความร้อนของตู้ชุมสายโทรศัพท์

1.5.3 หาวิธีการวัดความเร็วรอบของพัดลมในระบบระบายความร้อนของตู้ชุมสายโทรศัพท์

1.5.4 ออกแบบและสร้างการส่งข้อมูลเข้าสู่ Web Application

1.5.5 ทดลองการใช้โดยการสมมุติในการทำงานจากการเก็บข้อมูล

1.6 โครงสร้างของรายงาน

1.6.1 ผลที่ได้จากการค้นคว้าทฤษฎี ที่เกี่ยวข้อง การสร้างและทดสอบได้นำมารายงานในรายงานฉบับนี้ซึ่งมีเนื้อหาแบ่งออกเป็น 5 บท แต่ละบทจะมีเนื้อหา ดังนี้

1.6.2 บทที่ 1 กล่าวถึง ความเป็นมา และจุดประสงค์ของโครงงานนี้

บทที่ 2 กล่าวถึง ทฤษฎีต่างๆที่เกี่ยวข้องกับโครงงานนี้

บทที่ 3 กล่าวถึง การออกแบบ

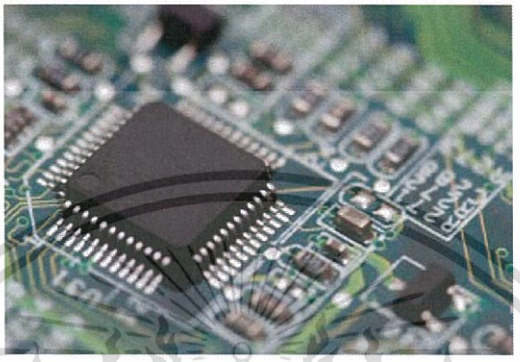
บทที่ 4 กล่าวถึง ผลการทดสอบ

บทที่ 5 กล่าวถึง บทสรุปของโครงงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่2
ทฤษฎีและหลักการที่เกี่ยวข้อง

2.1 Microcontroller



รูปที่ 2.1 Microcontroller

ไมโครคอนโทรลเลอร์ (Microcontroller มักย่อว่า μC, μC หรือ MCU) คือ อุปกรณ์ควบคุมขนาดเล็ก ซึ่งบรรจุความสามารถที่คล้ายคลึงกับระบบคอมพิวเตอร์ โดยในไมโครคอนโทรลเลอร์ได้รวมเอา ซีพียู , หน่วยความจำ และ พอร์ต ซึ่งเป็นส่วนประกอบหลักสำคัญของระบบคอมพิวเตอร์เข้าไว้ด้วยกัน โดยทำการบรรจุเข้าไว้ในตัวถังเดียวกัน

โครงสร้างโดยทั่วไปของไมโครคอนโทรลเลอร์นั้น สามารถแบ่งออกมาได้เป็น 5 ส่วนใหญ่ๆ ดังต่อไปนี้

- หน่วยประมวลผลกลางหรือซีพียู (CPU : Central Processing Unit)
- หน่วยความจำ (Memory) สามารถแบ่งออกเป็น 2 ส่วน คือ หน่วยความจำที่มีไว้สำหรับเก็บโปรแกรมหลัก (Program Memory) เปรียบเสมือนฮาร์ดดิสก์ของเครื่องคอมพิวเตอร์ตั้งโต๊ะ คือข้อมูลใดๆ ที่ถูกเก็บไว้ในนี้จะไม่สูญหายไปแม้ไม่มีไฟเลี้ยง อีกส่วนหนึ่งคือหน่วยความจำข้อมูล(DataMemory)ใช้เป็นเหมือนกระดาษทดในการคำนวณของซีพียู และเป็นที่พักข้อมูลชั่วคราวขณะทำงาน แต่หากไม่มีไฟเลี้ยง ข้อมูลก็จะหายไปคล้ายกับหน่วยความแรม (RAM) ในเครื่องคอมพิวเตอร์ทั่วๆ ไป แต่สำหรับไมโครคอนโทรลเลอร์สมัยใหม่ หน่วยความจำข้อมูลจะมีทั้งที่เป็นหน่วยความจำแรมเป็นหน่วยความจำแรมซึ่งข้อมูลจะหายไปเมื่อไม่มีไฟเลี้ยง และเป็น อีอีพรอม (EEPROM : Erasable Electrically Read-Only Memory) ซึ่งสามารถเก็บข้อมูลได้แม้ไม่มีไฟเลี้ยงก็ตาม
- ส่วนติดต่อกับอุปกรณ์ภายนอก หรือพอร์ต (Port) มี 2 ลักษณะคือ พอร์ตอินพุต (Input Port) และพอร์ตส่งสัญญาณหรือพอร์ตเอาต์พุต (Output Port) ส่วนนี้จะใช้ในการเชื่อมต่อกับอุปกรณ์ภายนอก ถือว่าเป็นส่วนที่สำคัญมาก ใช้ร่วมกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระหว่างพอร์ตอินพุต เพื่อรับสัญญาณ อาจจะด้วยการกดสวิตช์ เพื่อนำไปประมวลผลและส่งไปพอร์ตเอาต์พุต เพื่อแสดงผลเช่น การติดสว่างของหลอดไฟ เป็นต้น

- ช่องทางเดินของสัญญาณ หรือบัส (BUS) คือเส้นทางการแลกเปลี่ยนสัญญาณข้อมูลระหว่าง ซีพียู หน่วยความจำและพอร์ต เป็นลักษณะของสายสัญญาณจำนวนมากอยู่ภายในตัวไมโครคอนโทรลเลอร์ โดยแบ่งเป็นบัสข้อมูล (Data Bus) , บัสแอดเดรส (Address Bus) และบัสควบคุม (Control Bus)

- วงจรกำเนิดสัญญาณนาฬิกา นับเป็นส่วนประกอบที่สำคัญมากอีกส่วนหนึ่ง เนื่องจากการทำงานที่เกิดขึ้นในตัวไมโครคอนโทรลเลอร์ จะขึ้นอยู่กับการกำหนดจังหวะ หากสัญญาณนาฬิกามีความถี่สูง จังหวะการทำงานก็จะสามารถทำได้ถี่ขึ้นส่งผลให้ไมโครคอนโทรลเลอร์ตัวนั้น มีความเร็วในการประมวลผลสูงตามไปด้วย

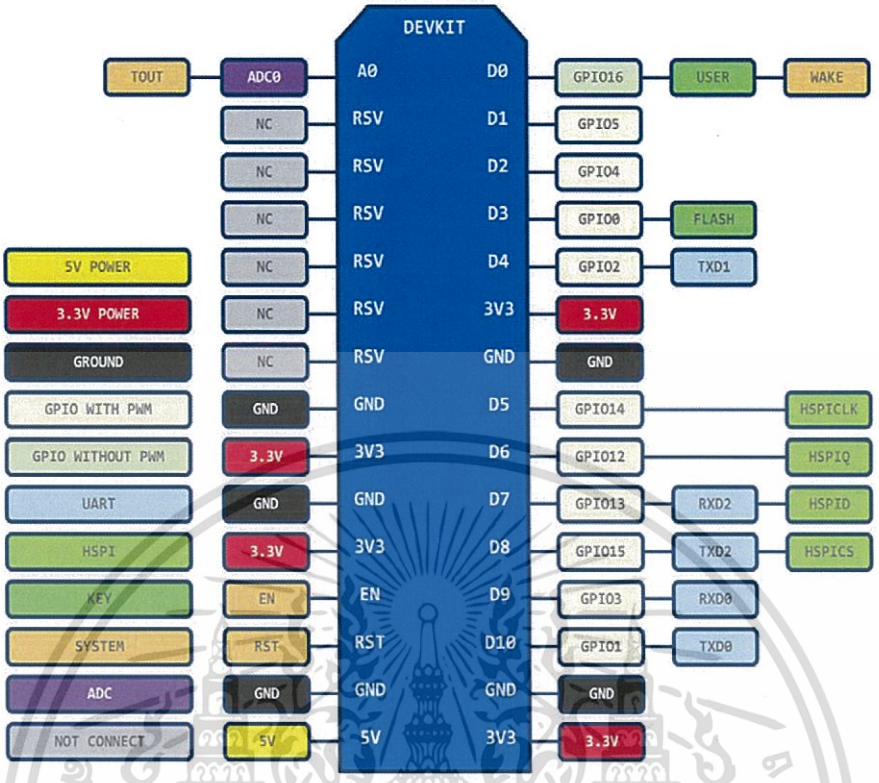
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 NodeMCU

รูปที่ 2.2 รูปบอร์ด NodeMCU

NodeMCU เป็นโมดูลที่ประกอบด้วย ESP8266-12 E มีเสาอากาศแบบ PCB Antenna เชื่อมต่อ header สำหรับขาสัญญาณต่างๆ 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 Lab CP2102 เชื่อมต่อเข้ากับเครื่องคอมพิวเตอร์เพื่อพัฒนาโปรแกรม สามารถติดตั้งเฟิร์มแวร์ Node MCU ได้ และยังมีขนาดของ PCB ที่เล็กลง สามารถใช้งานกับ breadboardได้ ผู้ใช้สามารถเลือกพัฒนาด้วยสคริปต์ LUA โดยใช้เฟิร์มแวร์ NodeMCU หรือใช้เป็นชุดพัฒนาด้วยโมดูล ESP8266 ก็ได้ซึ่งสามารถเขียนด้วย Arduino IDE ได้ โมดูลมี GPIO ให้ใช้ถึง 10 พอร์ต สามารถนำมาพัฒนาโปรเจ็คทางด้าน Internet of Things (IoTs) เชื่อมต่อกับอุปกรณ์อื่นๆ ตามต้องการ

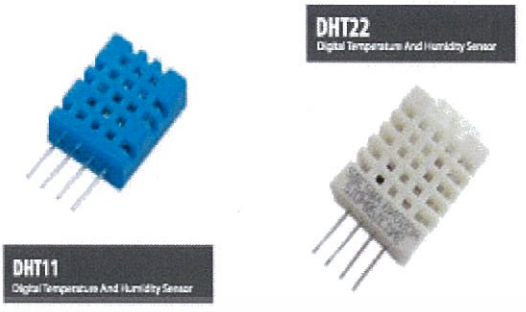
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.3 รายละเอียดหน้าที่การทำงานของขาต่างๆของ Node MCU

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 เซนเซอร์ตรวจวัดอุณหภูมิ (Temperature Sensor)



รูปที่ 2.4 Temperature Sensor (DHT11,DHT22)

เซนเซอร์ตรวจวัดอุณหภูมิเป็นการตรวจวัดอุณหภูมิใช้รูปแบบการเปลี่ยนแปลงระดับแรงดันไฟฟ้าจากสัญญาณอนาล็อกไปสู่ สัญญาณดิจิตอล โดยสัมพันธ์กับอุณหภูมิ มีรูปแบบใหญ่ ๆ ของ เซนเซอร์ อยู่ด้วยกัน 3 รูปแบบ คือ

- **Thermocouple** คือ อุปกรณ์วัดอุณหภูมิโดยใช้หลักการเปลี่ยนแปลงอุณหภูมิเป็นแรงเคลื่อนไฟฟ้า ทำมาจากโลหะตัวนำที่ต่างชนิดกัน 2 ตัวมาเชื่อมต่อปลายทั้งสองเข้าด้วยกัน ที่ปลายด้านหนึ่ง เรียกว่า"จุดอุณหภูมิ"ส่วนปลายอีกด้านหนึ่งปล่อยเปิดไว้ เรียกว่า"จุดอ้างอิง"หากที่จุดวัดอุณหภูมิและจุดอ้างอิงมีอุณหภูมิต่างกันก็จะทำให้มีการนำกระแส ในวงจร Thermocouple

รูปที่ 2.5 Thermocouple

- **Resistance Temperature Detector (RTD)** คือ ตัวเซ็นเซอร์อุณหภูมิที่ใช้หลักการเปลี่ยนแปลงค่าความต้านทานของโลหะ ซึ่งค่าความต้านทานดังกล่าวจะมีค่าเพิ่มตามอุณหภูมิ ความต้านทานของโลหะที่เพิ่มขึ้นนี้ เรียกว่า “สัมประสิทธิ์การเปลี่ยนแปลงอุณหภูมิแบบบวก”นิยมนำไปใช้ในการวัดอุณหภูมิในช่วง -270 to 850 °C. วัสดุที่นำมาใช้จะเป็นโลหะที่มีความต้านทานจำเพาะต่ำ เช่น แพลทินัม, ทังสเตน และ นิกเกิล

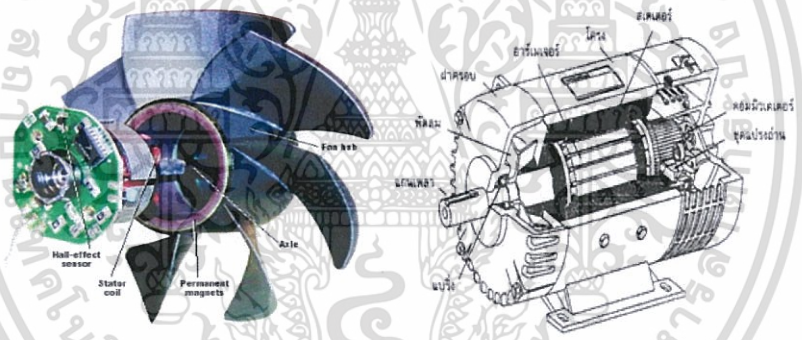
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **Thermistor** เป็น อุปกรณ์ความต้านทานชนิดที่สามารถเปลี่ยนค่าความต้านทานเมื่อได้รับความร้อน โดยที่ค่าความต้านทานจะเปลี่ยนแปลงแบบไม่เป็นเชิงเส้น กับอุณหภูมิ แบ่งเป็น 2 ลักษณะ คือ

  Positive Temperature Comitial (PTC) เป็น ชนิดที่ปกติจะมีค่าความต้านทานต่ำ เมื่อได้รับความร้อนจะทำให้มีค่าความต้านทานสูงขึ้นตามลำดับอุณหภูมิ นำไปใช้ตรวจสอบระดับความร้อน หรือทำให้เกิดความร้อนขึ้นเพื่อควบคุมการจ่ายแรงดันไฟฟ้าให้กับขดลวด เช่น วงจรล้างสนามแม่เหล็กอัตโนมัติของเครื่องรับโทรทัศน์สี (Degaussing coil) เป็นต้น

  Negative Temperature Comitial (NTC) เป็นชนิด ที่ปกติจะมีความต้านทานสูงเมื่อได้รับความร้อน ค่าความต้านทานจะต่ำลง ใช้งานด้านการตรวจสอบความร้อนเพื่อควบคุมระดับการทำงาน เช่น ในวงจรขยายเสียงที่ดีใช้ตรวจจับความร้อนที่เกิดจากการทำงานแล้วป้อนกลับไปลด การทำงานของวงจรให้น้อยลง เพื่ออุปกรณ์หลักจะไม่เกิดความร้อนมากจนเกินไป

2.4 มอเตอร์ไฟฟ้ากระแสตรง (DC Motor)



รูปที่ 2.6 DC Fan motor

มอเตอร์ไฟฟ้าเป็นอุปกรณ์ที่นิยมใช้กันอย่างแพร่หลายในโรงงานต่างเป็นอุปกรณ์ที่ใช้ควบคุมเครื่องจักรกลต่างๆในงานอุตสาหกรรมมอเตอร์มีหลายแบบหลายชนิดที่ใช้ให้เหมาะสมกับงานดังนั้นเราจึงต้องทราบถึงความหมายและชนิดของมอเตอร์ไฟฟ้าตลอดคุณสมบัติการใช้งานของมอเตอร์แต่ละชนิดเพื่อให้เกิดประสิทธิภาพสูงสุดในการใช้งานของมอเตอร์นั้นๆและสามารถเลือกใช้งานให้เหมาะสมหรืองานอื่นที่เกี่ยวข้องได้

มอเตอร์ไฟฟ้ากระแสตรง เป็นต้นกำลังขับเคลื่อนที่สำคัญอย่างหนึ่งในโรงงานอุตสาหกรรม เพราะมีคุณสมบัติที่ดีเด่นในด้านการปรับความเร็วได้ตั้งแต่ความเร็วต่ำสุดจนถึงสูงสุด นิยมใช้กันมากในโรงงานอุตสาหกรรม เช่นโรงงานทอผ้า โรงงานเส้นใยโพลีเอสเตอร์ โรงงานถลุงโลหะหรือให้ เป็นต้นกำลังในการขับเคลื่อนรถไฟฟ้า เป็นต้นในการศึกษาเกี่ยวกับมอเตอร์ไฟฟ้ากระแสตรงจึงควรรู้จักอุปกรณ์ต่างๆ ของมอเตอร์ไฟฟ้ากระแสตรงและเข้าใจถึงหลักการทำงานของมอเตอร์ไฟฟ้ากระแสตรงแบบต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนประกอบของมอเตอร์ไฟฟ้ากระแสตรง

มอเตอร์ไฟฟ้ากระแสตรงที่ส่วนประกอบที่สำคัญ 2 ส่วนดังนี้

2.4.1 ส่วนที่อยู่กับที่หรือที่เรียกว่าสเตเตอร์ (Stator) ประกอบด้วย

2.4.1.1 เฟรมหรือโยค (Frame Or Yoke) เป็นโครงภายนอกทำหน้าที่เป็นทางเดินของเส้นแรงแม่เหล็กจากขั้วเหนือไปขั้วใต้ให้ครบวงจรและยึดส่วนประกอบอื่นๆให้แข็งแรงทำด้วยเหล็กหล่อหรือเหล็กแผ่นหนาม้วนเป็นรูปทรงกระบอก

รูปที่ 2.7 Stator

2.4.1.2 ขั้วแม่เหล็ก (Pole) แกนขั้ว (Pole Core) จะทำด้วยแผ่นเหล็กบางๆ กั้นด้วยฉนวนประกอบกันเป็นแท่งยึดติดกับเฟรม ส่วนปลายที่ทำเป็นรูปโค้งนั้นเพื่อโค้งรับรูปกลมของตัวโรเตอร์เรียกว่าขั้วแม่เหล็ก (Pole Shoes) มีวัตถุประสงค์ให้ขั้วแม่เหล็กและโรเตอร์ใกล้ชิดกันมากที่สุดเพื่อให้เกิดช่องอากาศน้อยที่สุด เพื่อให้เกิดช่องอากาศน้อยที่สุดจะมีผลให้เส้นแรงแม่เหล็กจากขั้วแม่เหล็กจากขั้วแม่เหล็กผ่านไปยังโรเตอร์มากที่สุดแล้วทำให้เกิดแรงบิดหรือกำลังบิดของโรเตอร์มากเป็นการทำให้มอเตอร์มีกำลังหมุน (Torque)

2.4.1.3 ขดลวดสนามแม่เหล็ก (Field Coil) จะพันอยู่รอบๆแกนขั้วแม่เหล็กขดลวดนี้ทำหน้าที่รับกระแสจากภายนอกเพื่อสร้างเส้นแรงแม่เหล็กให้เกิดขึ้น และเส้นแรงแม่เหล็กนี้จะเกิดการหักล้างและเสริมกันกับสนามแม่เหล็กของอาเมเจอร์ทำให้เกิดแรงบิดขึ้น

2.4.2 ตัวหมุน (Rotor)

ตัวหมุนหรือเรียกว่าโรเตอร์ตัวหมุนนี้ทำให้เกิดกำลังงานมีแกนวางอยู่ในตลับลูกปืน (Ball Bearing) ซึ่งประกอบอยู่ในแผ่นปิดหัวท้ายของมอเตอร์

รูปที่ 2.8 Rotor

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวโรเตอร์ประกอบด้วย 4 ส่วนด้วยกัน คือ

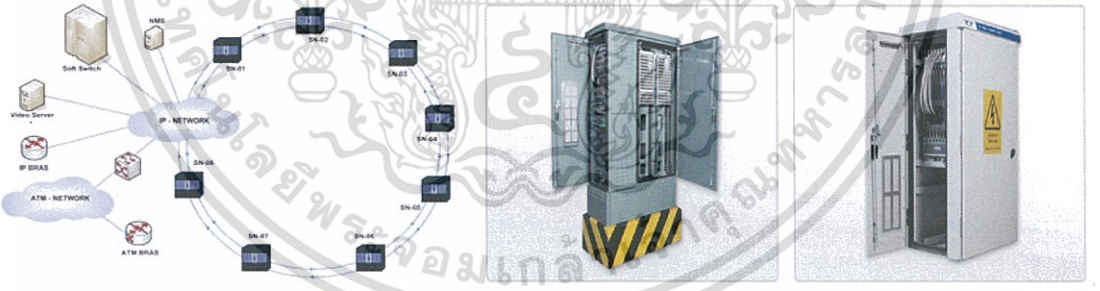
2.4.2.1 แกนเพลา (Shaft) เป็นตัวสำหรับยึดคอมมิวเตเตอร์และยึดแกนเหล็กอาร์มาเจอร์ (Armature Core) ประกอบเป็นตัวโรเตอร์แกนเพลานี้จะวางอยู่บนแบริ่งเพื่อบังคับให้หมุนอยู่ในแนวนิ่งไม่มีการสั่นสะเทือนได้

2.4.2.2 แกนเหล็กอาร์มาเจอร์ (Armature Core) ทำด้วยแผ่นเหล็กบางอาบฉนวนเป็นที่สำหรับพันขดลวดอาร์มาเจอร์ซึ่งสร้างแรงบิด(Torque)

2.4.2.3 คอมมิวเตเตอร์ (Commutator) ทำด้วยทองแดงออกแบบเป็นซี่แต่ละซี่มีฉนวนไมก้า คั่นระหว่างซี่ของคอมมิวเตเตอร์ส่วนหัวซี่ของคอมมิวเตเตอร์จะมีร่องสำหรับใส่ปลายสายของขดลวดอาร์มาเจอร์ตัวคอมมิวเตเตอร์นี้อัดแน่นติดกับแกนเพลาเป็นรูปกลมทรงกระบอก มีหน้าที่สัมผัสกับแปรงถ่าน(CarbonBrushes) เพื่อรับกระแสจากสายป้อนเข้าไปยัง ขดลวดอาร์มาเจอร์เพื่อสร้างเส้นแรงแม่เหล็กอีกส่วนหนึ่งให้เกิดการหักล้างและเสริมกันกับเส้นแรงแม่เหล็กอีกส่วนซึ่งเกิดจากขดลวดขั้วแม่เหล็ก ดังกล่าวมาแล้วเรียกว่าปฏิกิริยามอเตอร์(Motor action)

2.4.2.4 ขดลวดอาร์มาเจอร์ (Armature Winding) เป็นขดลวดพันอยู่ในร่องสลอท (Slot) ของแกนอาร์มาเจอร์ขนาดของลวดจะเล็กหรือใหญ่และจำนวนรอบจะมากหรือน้อยนั้นขึ้นอยู่กับการออกแบบของตัวโรเตอร์ชนิดนั้นๆ เพื่อที่จะให้เหมาะสมกับงานต่างๆ

2.5 อุปกรณ์ตู้ชุมสายโทรศัพท์และอินเตอร์เน็ต (MSAN Smart Node)



รูปที่ 2.9 ระบบเครือข่ายและอุปกรณ์ตู้ชุมสายโทรศัพท์และอินเตอร์เน็ต(MSAN)

อุปกรณ์ชุมสาย MSAN เป็นอุปกรณ์หลักเพื่อให้บริการ Narrowband และ Broadband ทุกประเภทภายใต้ชุมสายเดียว เพื่อตอบสนองลูกค้าที่มีความต้องการที่หลากหลาย ซึ่งมีทั้งการให้บริการผ่านสายสัญญาณแบบทองแดง และ สายสัญญาณแบบ OPTICAL Fiber และยังสามารถให้บริการ FTTX ได้อีกด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

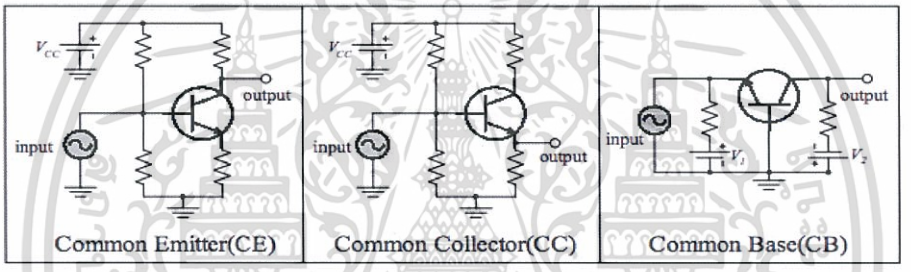
2.6 ทฤษฎีที่เกี่ยวข้อง

2.6.1 วงจรขยายสัญญาณโดยใช้ทรานซิสเตอร์แบบพื้นฐาน (Basic transistor amplifier configuration)

เนื่องจากทรานซิสเตอร์เป็นอุปกรณ์ที่มีขั้ว 3 ขั้ว ดังนั้นจึงสามารถต่อวงจรขยายได้3 แบบ โดยการระบุชื่อของวงจรนั้นบริเวณที่ขั้วใดต่อลง ground หรือเป็นขา common ซึ่งมีชื่อเรียกว่า

1. วงจรขยายอิมิตเตอร์ร่วม (Common emitter , CE)
2. วงจรขยายคอลเล็กเตอร์ร่วม (Common collector , CC)
3. วงจรขยายเบสร่วม (Common base , CB)

ชื่อวงจรขยาย	ขั้ว Input	ขั้ว Output
Common Emitter [CE]	Base [B]	Collector [C]
Common Collector [CC]	Base [B]	Emitter [E]
Common Base [CB]	Emitter [E]	Collector [C]



รูปที่ 2.10 แสดงลักษณะของวงจรขยาย BJT 3 ชนิด

จะสังเกตได้ว่า จะไม่มีการนำเอาขั้วคอลเลคเตอร์ไปเป็นขั้ว input และนำเอาขั้วเบส ไปเป็นขั้ว output เนื่องจาก ข้อจำกัดทางด้านโครงสร้างของ BJT [เฉพาะทรานซิสเตอร์แบบ npn เท่านั้น]

2.6.1.1 วงจรขยายสัญญาณอิมิตเตอร์ร่วม (Common Emitter , CE)

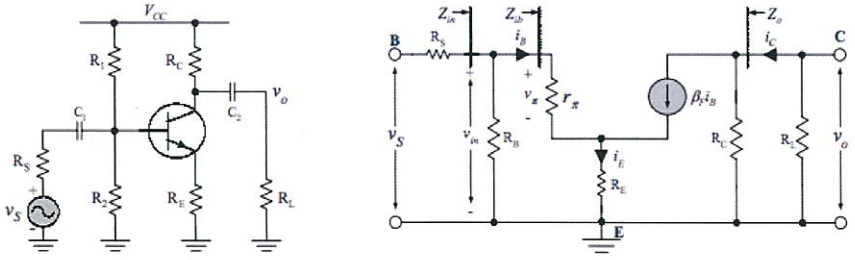
วงจรขยายสัญญาณอิมิตเตอร์ร่วม หรือเรียกอีกอย่างว่าวงจรขยายกลับเฟส(inverting amplifier)มีลักษณะวงจรแสดงดังรูป 2.11 ลักษณะพิเศษของวงจรคือสัญญาณอินพุตและสัญญาณเอ้าพุตจำมีความต่างเฟส กัน 180 องศา หรือกลับ เฟสกัน จากรูป 2.11 ถ้ากำหนดให้ VCC มีค่าเป็นศูนย์แต่R1 และ R2 ยังคงเป็นส่วน หนึ่งของอินพุตเราจึง

กำหนดให้ $$R_B = R_1 // R_2 = \frac{R_1 R_2}{R_1 + R_2}$$

จากนั้นตั้งสมการ KVL ทางด้านอินพุตก็จะได้ $$v_{in} = i_B r_\pi + i_E R_E = i_B r_\pi + (\beta_F + 1) i_B R_E$$

$$v_o = -\beta_F i_B [R_C // R_L]$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ที่ 2.11 (รูปซ้าย) วงจรขยาย common emitter (รูปขวา) เป็นวงจรเทียบเท่า

จะได้ $$A_v = \frac{-\beta_F [R_C \,//\, R_L]}{r_\pi + (\beta_F + 1)R_E}$$

จากสมการด้านบน ถ้ากำหนดให้ βF มีค่าสูงมากๆ ก็อาจประมาณได้ว่า

$$A_v = -\frac{(R_C \,//\, R_L)}{(1/g_m + R_E)}$$

การหาค่า input impedance สามารถหาได้จากสมการ $A_v = \frac{v_{in}}{i_{in}}$

ตั้งสมการหาค่าได้จาก $v_{in} = i_B r_\pi + i_E R_E$ หรือ

$$v_{in} = i_B r_\pi + (\beta_F + 1) i_B R_E = [r_\pi + (\beta_F + 1)R_E] i_B$$

$$\frac{v_{in}}{i_B} = r_\pi + (\beta_F + 1)R_E$$

จากรูป ก็จะได้ค่า input impedance , Zin เป็น $Z_{in} = R_B \,//\, [r_\pi + (\beta_F + 1)R_E]$

ในทำนองเดียวกันก็จะได้ค่า output impedance, ZO เป็น $Z_O = R_C$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6.1.2 วงจรขยายแบบอิมิตเตอร์ร่วมที่มีการ Bypass(Bypassed Common Emitter)

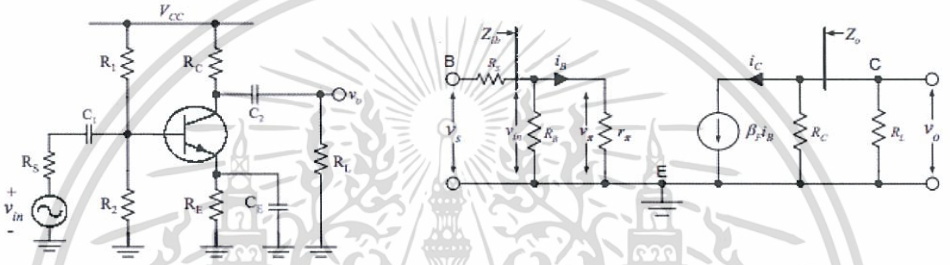
เป็นวงจรขยาย CE ที่มีการนำเอาตัวเก็บประจุ C มาต่อขนานกับ RE ซึ่งจะทำให้วงจรมีค่า input impedance และ voltage gain ที่สูงขึ้นกว่าเดิมมากแต่ไม่ทำให้ไบอัสไฟตรงเปลี่ยนแปลงไป ดังรูป 2.12 จากรูป 2.12

ก็ยังพบว่า $Z_{ib} = R_B // r_\pi$

$Z_o = R_C$

และจากวงจรยังพบว่า $v_o = -\beta_F i_B (R_C // R_L)$

และ $v_{in} = i_B r_\pi$



รูปที่ 2.12 (รูปซ้าย) วงจรขยาย bypassed common emitter (รูปขวา) เป็นวงจรเทียบเท่า

จะได้ $$A_v = \frac{-\beta_F i_B [R_C // R_L]}{i_B r_\pi} = \frac{-\beta_F [R_C // R_L]}{r_\pi} = -g_m [R_C // R_L]$$

สำหรับการหาค่าอัตราขยายกระแส หรือ current gain ก็ถือเป็นพารามิเตอร์อีกตัวหนึ่งที่มีความสำคัญเช่นเดียวกัน เป็นอัตราส่วนระหว่างกระแสเอ้าพุตกับกระแสอินพุต

ดังสมการ $$A_i = \frac{i_o}{i_{in}}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากวงจรในรูป 2.12 จะหาค่า current gain ได้จากกฎการแบ่งกระแส

จะได้ $i_o = \beta_F i_B$ จะได้ $\beta_F = \frac{i_o}{i_B}$

$i_B = i_{in}\left(\frac{R_B}{R_B + r_\pi}\right)$ จะได้ $\frac{i_B}{i_{in}} = \frac{R_B}{R_B + r_\pi}$

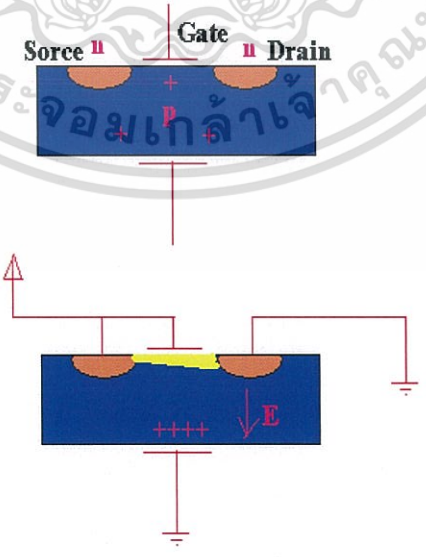
$$\frac{i_o}{i_{in}} = \frac{i_o}{i_B} \cdot \frac{i_B}{i_{in}} = \frac{R_B \beta_F}{R_B + r_\pi}$$

$$A_i = \frac{i_o}{i_{in}} = \frac{\beta_F}{1 + \left(\frac{r_\pi}{R_B}\right)}$$

2.6.2 MOSFET

ในทางดิจิตอลถูกมองว่าเป็นสวิตช์ โดย nMOS จะเป็นสวิตช์ที่เมื่อสัญญาณเข้าเป็น "1" สวิตช์ก็จะปิด ถ้าไม่สวิตช์ก็ยังเปิดอยู่ (normal opened switch) ส่วน pMOS จะเป็นสวิตช์ที่เมื่อสัญญาณเข้าเป็น "1" สวิตช์ก็จะเปิด ถ้าไม่สวิตช์ก็จะปิดอยู่ (normal closed switch) และสัญลักษณ์ทั่วไปจะมีสามขา ขากลางเป็น gate ส่วนอีกสองขาคือ source และ drain โดยใช้ใน nMOS เป็นหลักเพื่อสื่อสัญลักษณ์เดียวกับทรานซิสเตอร์ทั่วไปคือ ไฟขา base ไหล ขา Collector จะต่อกับ Emittor ส่วน pMOS ก็จะใส่ bubble ที่ขา gate

การทำงานของ MOSFET



รูปที่ 2.13 การทำงานของ MOSFET

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- nMOS เมื่อปล่อยความต่างศักย์สูง จะเกิดสนามไฟฟ้าในทิศลงอย่างแรง โฮลใน p-type จะถูกผลักลงมาอยู่ด้านล่าง (ตามรูปที่ประกอบข้างบน) ประกอบกับมีอิเล็กตรอนอิสระบางส่วนถูกดูดขึ้นไปด้านบน ส่งผลให้บริเวณด้านบนมีอิเล็กตรอนอิสระมากจนเป็น n-type ได้เรียกว่า channel สัญญาณไฟฟ้าก็จะไหลผ่านช่วง channel นี้ซึ่งเป็น n-type เหมือนกับ drain และ source ได้โดยใช้อิเล็กตรอนอิสระเป็นพาหะ

- pMOS จะทำงานกลับกับ nMOS โดยเมื่อปล่อยความต่างศักย์ต่ำ (โดยมากมักจะติดลบ) จะเกิดสนามไฟฟ้าในทิศขึ้นอย่างแรง อิเล็กตรอนอิสระใน n-type จะถูกผลักลงมาอยู่ด้านล่าง ประกอบกับมีโฮลบางส่วนถูกดูดขึ้นไปด้านบน ส่งผลให้บริเวณด้านบนมีโฮลมากจนเป็น p-type ได้เรียกว่า channel สัญญาณไฟฟ้าก็จะไหลผ่านช่วง channel นี้ซึ่งเป็น p-type เหมือนกับ drain และ sorce ได้โดยใช้โฮลเป็นพาหะ

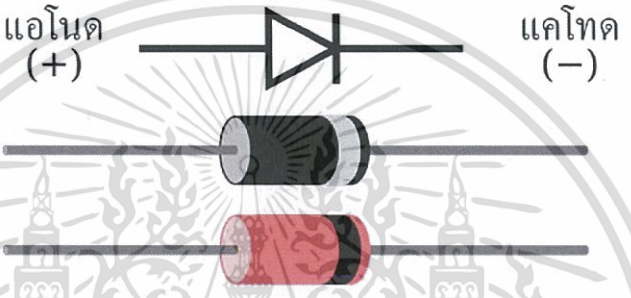
2.6.3 ไดโอด (Diode)

เป็นชิ้นส่วนอิเล็กทรอนิกส์ชนิดสองขั้วคือขั้ว p และขั้ว n ที่ออกแบบและควบคุมทิศทางการไหลของประจุไฟฟ้า มันจะยอมให้กระแสไฟฟ้าไหลในทิศทางเดียว และกั้นการไหลในทิศทางตรงกันข้าม เมื่อกล่าวถึงไดโอด มักจะหมายถึงไดโอดที่ทำมาจากสารกึ่งตัวนำ (Semiconductor diode) ซึ่งก็คือผลึกของสารกึ่งตัวนำที่ต่อกันได้ขั้วทางไฟฟ้าสองขั้ว ส่วนไดโอดแบบหลอดสุญญากาศ (Vacuum tube diode) ถูกใช้เฉพาะทางในเทคโนโลยีไฟฟ้าแรงสูงบางประเภท เป็นหลอดสุญญากาศที่ประกอบด้วยขั้วอิเล็กโทรดสองขั้ว ซึ่งจะคือแผ่นตัวนำ (plate) และแคโทด (cathode) ส่วนใหญ่เราจะใช้ไดโอดในการยอมให้กระแสไปในทิศทางเดียว โดยยอมให้กระแสไฟในทางใดทางหนึ่ง ส่วนกระแสที่ไหลทิศทางตรงข้ามกันจะถูกกั้น ดังนั้นจึงอาจถือว่าไดโอดเป็นวาล์วตรวจสอบแบบอิเล็กทรอนิกส์อย่างหนึ่ง ซึ่งนับเป็นประโยชน์อย่างมากในวงจรอิเล็กทรอนิกส์ เช่น ใช้เป็นตัวเรียงกระแสไฟฟ้าในวงจรแหล่งจ่ายไฟ เป็นต้น อย่างไรก็ตามไดโอมีความสามารถมากกว่าการเป็นอุปกรณ์ที่ใช้เปิด-ปิดกระแสง่าย ๆ ไดโอดมีคุณลักษณะทางไฟฟ้าที่ไม่เป็นเชิงเส้น ดังนั้นมันยังสามารถปรับปรุงโดยการปรับเปลี่ยนโครงสร้างของพวกมันที่เรียกว่ารอยต่อ p-n มันถูกนำไปใช้ประโยชน์ในงานที่มีวัตถุประสงค์พิเศษ นั่นทำให้ไดโอดมีรูปแบบการทำงานได้หลากหลายรูปแบบ ยกตัวอย่างเช่น ซีเนอร์ไดโอด เป็นไดโอดชนิดพิเศษที่ทำหน้าที่รักษาระดับแรงดันให้คงที่ วาริแอกไดโอดใช้ในการปรับแต่งสัญญาณในเครื่องรับวิทยุและโทรทัศน์ ไดโอดอุโมงค์หรือทันเนลไดโอดใช้ในการสร้างสัญญาณความถี่วิทยุ และไดโอดเปล่งแสงเป็นอุปกรณ์ที่สร้างแสงขึ้น ไดโอดอุโมงค์มีความน่าสนใจตรงที่มันจะมีค่าความต้านทานติดลบ ซึ่งเป็นประโยชน์มากเมื่อใช้ในวงจรบางประเภท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไดโอดตัวแรกเป็นอุปกรณ์หลอดสุญญากาศ โดยไดโอดแบบสารกึ่งตัวนำตัวแรกถูกค้นพบจากการทดสอบความสามารถในการเรียงกระแสของผลึกโดยคาร์ล เฟอร์ดินานด์ บรวน นักฟิสิกส์ชาวเยอรมัน ในปี พ.ศ. 2417 เรียกว่า cat's whisker diodes และได้ถูกพัฒนาในปี พ.ศ. 2449 โดยทำไดโอดมากผลึกแร่กาลีนา แต่ทุกวันนี้ไดโอดที่ใช้ทั่วไปผลิตมาจากสารกึ่งตัวนำ เช่น ซิลิกอน หรือ เจอร์เมเนียม

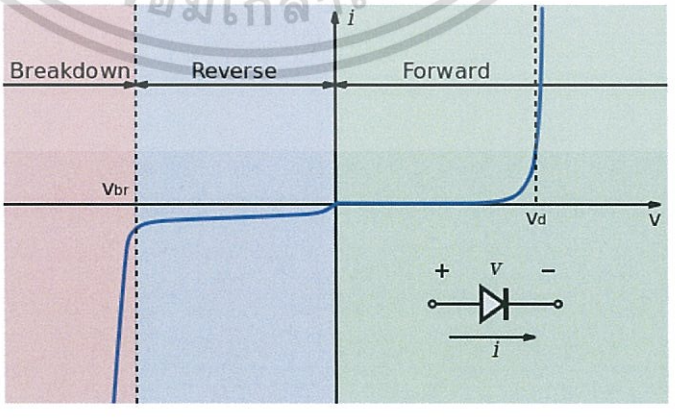
ไดโอดเป็นอุปกรณ์ที่ทำจากสารกึ่งตัวนำ p - n สามารถควบคุมให้กระแสไฟฟ้าจากภายนอกไหลผ่านตัวมันได้ทิศทางเดียว ไดโอดประกอบด้วยขั้ว 2 ขั้ว คือ แอโนด (Anode; A) ซึ่งต่ออยู่กับสารกึ่งตัวนำชนิด p และ แคโธด (Cathode; K) ซึ่งต่ออยู่กับสารกึ่งตัวนำชนิด n



รูปที่ 2.14 diode และสัญญาลักษณ์ และ ขั้วของ Diode

2.6.3.1 คุณลักษณะเฉพาะของกระแสและแรงดัน

พฤติกรรมของไดโอดแบบสารกึ่งตัวนำในวงจรจะก่อให้เกิดคุณลักษณะเฉพาะของกระแสและแรงดัน(current-voltage characteristic)หรือเรียกว่ากราฟ I-V รูป 2.6.6 รูปร่างของเส้นโค้งถูกกำหนดจากส่งผ่านประจุผ่านเขตปลอดพาหะ (depletion region หรือ depletion layer) ซึ่งอยู่ใยรอยต่อ p-n



รูปที่ 2.15 พฤติกรรมของไดโอด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6.3.2 พฤติกรรมของสัญญาณขนาดเล็ก

ในการออกแบบวงจร แบบจำลองของสัญญาณขนาดเล็กจากพฤติกรรมของไดโอดถูกนำมาใช้งานอยู่บ่อยครั้งแบบจำลองสัญญาณขนาดเล็ก(Small-signal model)เป็นเทคนิคการวิเคราะห์ทางวิศวกรรมไฟฟ้าที่อาศัยการประมาณพฤติกรรมของอุปกรณ์ทางไฟฟ้าที่ไม่มีความเป็นเชิงเส้น ด้วยสมการเชิงเส้น ความเป็นเชิงเส้นนี้ขึ้นอยู่กับจุดไบอัสกระแสตรง(DCbiaspoint)ของอุปกรณ์(นั่นก็คือระดับของแรงดัน/กระแส ที่แสดงออกเมื่อไม่มีสัญญาณที่ถูกนำมาใช้)และสามารถทำให้ถูกต้องได้ด้วยการมองที่จุดนี้อีกด้วย

2.7 ภาษาที่ใช้เขียนเพื่อออกแบบโปรแกรม

2.7.1 ภาษาซี (C)

ภาษาซี (C) เป็นภาษาโปรแกรมสำหรับวัตถุประสงค์ทั่วไป เริ่มพัฒนาขึ้นระหว่าง พ.ศ. 2512-2516 (ค.ศ. 1969-1973) โดยเดนนิส ริชชี่ (Denis Retchie) ที่เอทีแอนด์ทีเบลล์แล็บส์ (AT&T Bell Labs) ภาษาซีเป็นภาษาที่มีความยืดหยุ่นในการเขียนโปรแกรมและมีเครื่องมืออำนวยความสะดวกสำหรับการเขียนโปรแกรมเชิงโครงสร้างและอนุญาตให้มีขอบข่ายตัวแปร (scope) และการเรียกซ้ำ (recursion) ในขณะที่ระบบชนิดตัวแปร ก็ช่วยป้องกันการดำเนินการที่ไม่ตั้งใจหลายอย่าง เหมือนกับภาษาโปรแกรมเชิงคำสั่งส่วนใหญ่ในแบบแผนของภาษาอัลกอล การออกแบบของภาษาซีมี(construct) ที่โยงกับชุดคำสั่งเครื่องทั่วไปได้อย่างพอเพียง จึงทำให้ยังมีการใช้ในโปรแกรมประยุกต์ซึ่งแต่ก่อนลงรหัสเป็นภาษาแอสเซมบลี คือซอฟต์แวร์ระบบอันโดดเด่นอย่างระบบปฏิบัติการคอมพิวเตอร์ ยูนิกซ์ ภาษาซีเป็นภาษาโปรแกรมหนึ่งที่ใช้กันอย่างแพร่หลายมากที่สุดตลอดกาล และตัวแปลโปรแกรมของภาษาซีมีให้ใช้งานได้สำหรับสถาปัตยกรรมคอมพิวเตอร์และระบบปฏิบัติการต่าง ๆ เป็นส่วนมาก ภาษาหลายภาษาในยุคหลังได้หยิบยืมภาษาซีไปใช้ทั้งทางตรงและทางอ้อม ตัวอย่างเช่น ภาษาดี ภาษาโก ภาษารัสต์ ภาษาจาวา จาวาสคริปต์ ภาษาลิมโบ ภาษาแอลพีซี ภาษาซีชาร์ป ภาษาอ็อบเจกทีฟ-ซี ภาษาเพิร์ล ภาษาพีเอชพี ภาษาไพทอน ภาษาเวอริล็อก (ภาษาพรรณนาฮาร์ดแวร์) และซีเชลล์ของยูนิกซ์ ภาษาเหล่านี้ได้ดึงโครงสร้างการควบคุมและคุณลักษณะพื้นฐานอื่น ๆ มาจากภาษาซี ส่วนใหญ่มีวากยสัมพันธ์คล้ายคลึงกับภาษาซีเป็นอย่างมากโดยรวม (ยกเว้นภาษาไพทอนที่ต่างออกไปอย่างสิ้นเชิง) และตั้งใจที่จะผสานนิพจน์และข้อความสั่งที่จำแนกได้ของวากยสัมพันธ์ของภาษาซี ด้วยระบบชนิดตัวแปร ตัวแบบข้อมูล และอรรถศาสตร์ที่อาจแตกต่างกันโดยมูลฐาน ภาษาซีพลัสพลัสและภาษาอ็อบเจกทีฟ-ซีเดิมเกิดขึ้นในฐานะตัวแปลโปรแกรมที่สร้างรหัสภาษาซี ปัจจุบันภาษาซีพลัสพลัสแทบจะเป็นเซตใหญ่ของภาษาซี ในขณะที่ภาษาอ็อบเจกทีฟ-ซีก็เป็นเซตใหญ่อันเคร่งครัดของภาษาซี ก่อนที่จะมีมาตรฐานภาษาซีอย่างเป็นทางการผู้ใช้และผู้พัฒนาต่างก็เชื่อถือในข้อกำหนดอย่างไม่เป็นทางการในหนังสือที่เขียนโดยเดนนิส ริตชี และไบรอัน เคอร์นิกัน (Brian Kernighan) ภาษาซีรุ่นนั้นจึงเรียกกันโดยทั่วไปว่า ภาษาเคแอนด์อาร์ซี (K&RC) ต่อมา พ.ศ. 2532 สถาบันมาตรฐานแห่งชาติของสหรัฐอเมริกา (ANSI) ได้ตีพิมพ์มาตรฐานสำหรับภาษาซีขึ้นมา เรียกกันว่า ภาษาแอนซีซี (ANSIC) หรือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
143924

ภาษา ซี89 (C89) ในปีถัดมา องค์การระหว่างประเทศว่าด้วยการมาตรฐาน (ISO) ได้อนุมัติให้ข้อกำหนดเดียวกันนี้เป็นมาตรฐานสากล เรียกกันว่า ภาษา ซี90 (C90) ในเวลาต่อมาอีกองค์การฯ ก็ได้เผยแพร่ส่วนขยายมาตรฐานเพื่อรองรับสากลวิวัตน์ (internationalization) เมื่อ พ.ศ. 2538 และมาตรฐานที่ตรวจชำระใหม่เมื่อ พ.ศ.2542 เรียกกันว่า ภาษา ซี99 (C99) มาตรฐานรุ่นปัจจุบันก็ได้รับอนุมัติเมื่อเดือนธันวาคม พ.ศ.2554 เรียกกันว่า ภาษาซี11 (C11)

รูปที่ 2.16 สัญญาลักษณ์ภาษา C

2.7.2 ไพทอน (Python)

Python เป็นภาษาระดับสูงภาษาหนึ่ง ที่มีความสามารถสูงถูกสร้างขึ้นในปี 1989 โดย Guido van Rossum ซึ่งถูกพัฒนาขึ้นมาโดยไม่ยึดติดกับแพลตฟอร์มกล่าวคือสามารถรันภาษา Pythonได้ทั้งบนระบบ Unix, Linux, WindowsNT, Windows 2000, Windows XP หรือแม้แต่ระบบ FreeBSD อีกอย่างหนึ่งภาษาตัวนี้เป็นภาษาลักษณะOpenSource เหมือนย่าง PHP

2.7.2.1 ความสามารถของภาษา Python

ปัจจุบันภาษาที่ใช้ในการพัฒนา Web Application มีมากมายหลายภาษา อาทิเช่น ภาษา Perl, PHP, JAVA, ASP, Tcl, Python เป็นต้น สำหรับภาษา Python นับว่ายังใหม่ในวงการพัฒนาโปรแกรมบนเว็บ แต่ด้วยข้อดีหลายประการของภาษา Pythonทำให้มีผู้นิยมใช้มากขึ้นเรื่อยๆ ซึ่งพอสรุปข้อดีของภาษา Python ได้ดังนี้

2.7.2.1.1 ง่ายต่อการเรียนรู้ โดยภาษา Python มีโครงสร้างของภาษาไม่ซับซ้อนเข้าใจง่าย ซึ่งโครงสร้างภาษา Python จะคล้ายกับภาษา C มาก เพราะภาษา Python สร้างขึ้นมาโดยใช้ภาษา C ทำให้ผู้ที่คุ้นเคยภาษา C อยู่แล้วใช้งานภาษา Python ได้ไม่ยาก นอกจากนี้โดยตัวภาษาเองมีความยืดหยุ่นสูงทำให้การจัดการกับงานด้านข้อความ และ Text File ได้เป็นอย่างดี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7.2.1.2 ไม่ต้องเสียค่าใช้จ่ายใดๆทั้งสิ้นเพราะตัวภาษา Pythonอยู่ภายใต้ลิขสิทธิ์ GNU

2.7.2.1.3 ใช้ได้หลายแพลตฟอร์ม ในช่วงแรกภาษา Pythonถูกออกแบบใช้งานกับระบบ Unix อยู่ก็จริงแต่ในปัจจุบันได้มีการพัฒนาตัวแปลภาษา Python ให้สามารถใช้กับระบบปฏิบัติการอื่นๆอาทิเช่น Linux, Windows 95/98/ME, Windows NT, Windows 2000, OS/2

2.7.2.1.4 ภาษา Python ถูกสร้างขึ้นโดยได้รวบรวมเอาส่วนดีของภาษาต่างๆ เข้ามาไว้ด้วยกัน อาทิเช่น ภาษา C, C++, Java, Perl

2.7.2.1.5 ภาษา Python เป็นภาษาประเภท Server side Script คือการทำงานของภาษา Python จะทำงานด้านฝั่ง Server แล้วส่งผลลัพธ์กลับมายัง Client ทำให้มีความปลอดภัยสูง

2.7.2.1.6 ใช้พัฒนา Web Service โดยที่ภาษา Python สามารถนำมาพัฒนาเว็บเซอร์วิส รวมทั้งใช้บริหารการสร้างเว็บไซต์สำเร็จรูปที่เรียกว่า Content Management Framework (CMF) ตัวอย่าง CMF ที่มีชื่อเสียงมากและเบื้องหลังทำงานด้วย python คือ Plone

2.7.2.2 หลักการทำงานของภาษา Python

ตัวแปรภาษาคืออะไร เมื่อเราได้เขียนโค้ดขึ้นมาตามโครงสร้างของโปรแกรมภาษาใดก็ตาม และการจะให้โค้ดคำสั่งเหล่านั้นทำงานได้ก็จะต้องมีตัวแปรภาษามาจัดการแปลโค้ดคำสั่ง เพื่อให้ทำงานตามที่เราต้องการโดยลักษณะของตัวแปลภาษานั้นแบ่งได้ 2ประเภทใหญ่ ๆ คือ

2.7.2.2.1 คอมไพเลอร์ (Compiler) เป็นตัวแปลภาษาสำหรับภาษา C, C++, Pascal การทำงานก็คือจะตรวจสอบความผิดพลาดของโค้ดคำสั่งตั้งแต่ต้นจนจบก่อน หรือเรียกว่าการคอมไพล์ ถ้าไม่มีข้อผิดพลาดก็จะทำการแปลโค้ดคำสั่งของเราให้เป็นไฟล์นามสกุล .obj (object file) จากนั้นก็ทำการแปลไฟล์ .obj ให้เป็นไบนารีไฟล์ .exe เพื่อทำงานต่อไป ดังตัวอย่างการทำงานของคอมไพเลอร์ภาษา C

2.7.2.2.2 อินเตอร์พรีเตอร์ (Interperter) จะทำงานเป็นบรรทัดต่อบรรทัด คือ อ่านโค้ดคำสั่งมาบรรทัดหนึ่งแล้วก็ทำงานให้ผลออกมาเลย

รูปที่ 2.17 สัญญาลักษณ์ภาษา Python

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7.3 ภาษา PHP

ในช่วงแรกภาษาที่นิยมใช้งานบนระบบเครือข่ายคือภาษาHTML(Hypertext Markup Language) แต่ภาษา HTML มีลักษณะเป็น Static คือภาษาที่มีลักษณะของข้อมูลคงที่ ซึ่งไม่เพียงพอต่อความต้องการในปัจจุบันที่นิยมใช้ระบบเครือข่าย Internet เป็นศูนย์กลางในการติดต่อระหว่างกัน ทำให้ต้องการใช้เว็บไซต์ที่มีลักษณะเป็นแบบ Dynamic คือ เว็บไซต์ที่ข้อมูลสามารถเปลี่ยนแปลงได้โดยอัตโนมัติตามเงื่อนไขต่าง ๆ ที่ผู้เขียนเว็บไซต์เป็นผู้กำหนด และการควบคุมการทำงานเหล่านี้จะกระทำโดยโปรแกรมภาษาสคริปต์ เช่น ภาษา PHP ซึ่งเป็นภาษาหนึ่งที่ได้รับความนิยมเป็นอย่างมากในปัจจุบัน

PHP ถูกสร้างขึ้นในปี ค.ศ.1994 โดย Rasmus Lerdorf ต่อมามีผู้ให้ความสนใจเป็นจำนวนมาก จึงได้ออกเป็นแพ็คเกจ "Personal Home Page" ซึ่งเป็นที่มาของ PHP โดยภาษา PHP เป็นแบบ Server Side Script และเป็นOpen Sourceที่ผู้ใช้ทั่วไปสามารถดาวน์โหลด Source Code และโปรแกรมไปใช้ฟรี ได้ที่ http://www.php.net พอกลางปี ค.ศ.1995 เขาก็ได้พัฒนาตัวแปลภาษา PHP ขึ้นมาใหม่ โดยใช้ชื่อว่า PHP/FI เวอร์ชั่น 2 ซึ่งได้เพิ่มความสามารถในการรับข้อมูลที่ส่งมาจากฟอร์มของ HTML (จึงมีชื่อว่า FI หรือ Form Interpreter) นอกจากนั้นยังเพิ่มความสามารถในการติดต่อกับฐานข้อมูลอีกด้วย จึงทำให้ผู้คนเริ่มหันมาสนใจ PHP กันมากขึ้น ในปี 1997 มีผู้ร่วมพัฒนา PHP เพิ่มอีก2 คนคือ Zeev Suraski และ Andi Gutmans (กลุ่มที่เรียกตัวเองว่า Zend ซึ่งย่อมาจาก Zeev และ Andi) โดยได้แก้ไขข้อบกพร่องต่างๆ และเพิ่มเติมเครื่องมือให้มากขึ้น

2.7.3.1 โครงสร้างของภาษา PHP

ภาษา PHP มีลักษณะเป็น embedded script หมายความว่าเราสามารถฝังคำสั่ง PHPไว้ในเว็บเพจร่วมกับคำสั่ง(Tag)ของHTMLได้และสร้างไฟล์ที่มีนามสกุลเป็น .php, .php3 หรือ .php4 ซึ่งไวยากรณ์ที่ใช้ใน PHP เป็นการนำรูปแบบของภาษาต่างๆ มารวมกันได้แก่ C, Perl และ Java ทำให้ผู้ใช้ที่มีพื้นฐานของภาษาเหล่านี้อยู่แล้วสามารถศึกษา และใช้งานภาษานี้ได้ไม่ยาก

```
1   <html>
2   <head>
3   <title>Example 1 </title>
4   </head>
5   <body>
6   <?
7     echo"Hi, I'm a PHP script!";
8   ?>
9   </body>
10  </html>
```

รูปที่ 2.18 ตัวอย่างการเขียนภาษา PHP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากตัวอย่าง บรรทัดที่ 6 - 8 เป็นส่วนของสคริปต์ PHP ซึ่งเริ่มต้นด้วย <? ตามด้วยคำสั่งที่เรียกฟังก์ชั่นหรือข้อความ และปิดท้ายด้วย ?> สำหรับตัวอย่างนี้เป็นสคริปต์ที่แสดงข้อความว่า"Hi, I'm a PHP script" โดยใช้คำสั่ง echo ซึ่งเป็นคำสั่งที่ใช้ในการแสดงผลของภาษาสคริปต์ PHP เราสามารถฝังคำสั่ง PHP ไว้ในเว็บเพจหนึ่งๆ โดยเปิดและปิดด้วยแท็ก(Tag) ของ PHP กี่ครั้งก็ได้ ดังตัวอย่างต่อไปนี้

```
1   <html>
2   <head>
3   <title>Example 1 </title>
4   </head>
5   <body>
6   <table border=1>
7   <tr>
8   <td><? echo"PHP script block 1"; ?></td>
9   <td><? echo"PHP script block 2 "; ?></td>
10  </tr>
11  </table>
12  <?
13    echo"PHP script block 3 <br> ";
14    echo date("ขณะนี้เวลา H:i น.");
15  ?>
16  </body>
17  </html>
```

รูปที่ 2.19 ตัวอย่างการเขียนภาษา PHP

2.7.3.2 ความสามารถของภาษา PHP

2.7.3.2.1 เป็นภาษาที่มีลักษณะเป็นแบบOpensource ผู้ใช้สามารถ Download และนำ Sourcecode ของ PHP ไปใช้ได้โดยไม่เสียค่าใช้จ่าย

2.7.3.2.2 เป็นสคริปต์แบบ Server Side Script ดังนั้นจึงทำงานบนเว็บเซิร์ฟเวอร์ ไม่ส่งผลกับการทำงานของเครื่อง Client โดย PHP จะอ่านโค้ด และทำงานที่เซิร์ฟเวอร์ จากนั้นจึงส่งผลลัพธ์ที่ได้จากการประมวลผลมาที่เครื่องของผู้ใช้ในรูปแบบของ HTML ซึ่งโค้ดของ PHP นี้ผู้ใช้จะไม่สามารถมองเห็นได้

2.7.3.2.3 PHP สามารถทำงานได้ในระบบปฏิบัติการที่ต่างชนิดกัน เช่น Unix,Windows,MacOSหรือRiscOS อย่างมีประสิทธิภาพ เนื่องจาก PHP เป็นสคริปต์ที่ต้องทำงานบนเซิร์ฟเวอร์ดังนั้นคอมพิวเตอร์สำหรับเรียกใช้คำสั่ง PHPจึงจำเป็นต้องติดตั้งโปรแกรมเว็บเซิร์ฟเวอร์ไว้ด้วย เพื่อให้สามารถประมวลผล PHP ได้

2.7.3.2.4 PHP สามารถทำงานได้ในเว็บเซิร์ฟเวอร์หลายชนิด เช่น Personal Web Server(PWS), Apache ,OmniHttpd และ Internet Information Service(IIS) เป็นต้น

2.7.3.2.5 ภาษา PHP สนับสนุนการเขียนโปรแกรมเชิงวัตถุ (Object Oriented Programming)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7.3.2.6 PHP มีความสามารถในการทำงานร่วมกับระบบจัดการฐานข้อมูลที่หลากหลาย ซึ่งระบบจัดการฐานข้อมูลที่สนับสนุนการทำงานของ PHPเช่น Oracle,MySQL ,FilePro ,Solid ,FrontBase , mSQL และ MS SQL เป็นต้น

2.7.3.2.7 PHP อนุญาตให้ผู้ใช้สร้างเว็บไซต์ซึ่งทำงานผ่านโปรโตคอลชนิดต่างๆ ได้ เช่น LDAP, IMAP, SNMP, POP3 และ HTTP เป็นต้น

2.7.3.2.8 โค้ด PHP สามารถเขียน และอ่านในรูปแบบของ XML ได้

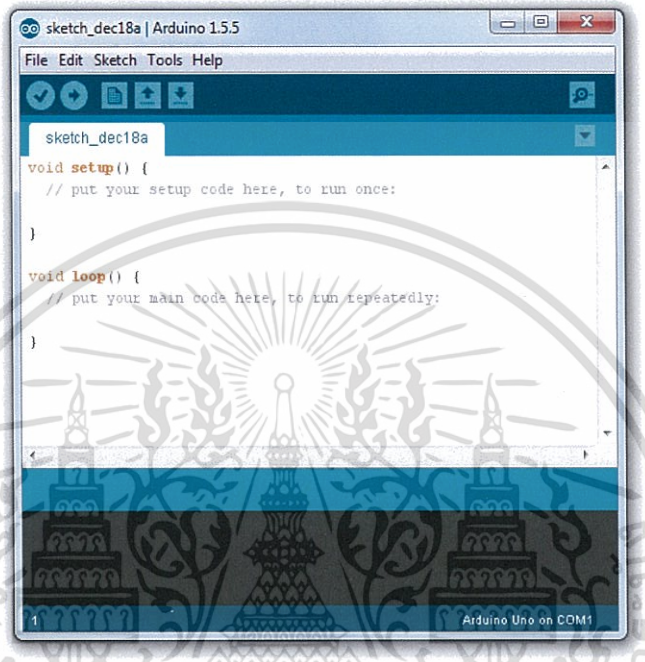
2.8 โปรแกรมที่ใช้สำหรับการออกแบบ

2.8.1 Arduino (อาดูยโน่)

เป็นระบบที่ใช้ในการพัฒนาอุปกรณ์อิเล็กทรอนิกส์ต้นแบบซึ่งออกแบบให้ใช้งานง่ายทั้งฮาร์ดแวร์ และซอฟแวร์ กลุ่มเป้าหมายผู้ใช้งานอาดูยโน่ ได้แก่ศิลปิน นักประดิษฐ์ นักออกแบบ ใช้ในงานอดิเรก หรือ ใครๆก็ตามที่สนใจในการประดิษฐ์นวัตกรรม งานสร้างสรรค์ มารู้จักกับที่มาที่ไปของ Arduino กันก่อน Ardiono มีผู้ริเริ่มเป็นชาวอิตาเลียน ดังนั้นจึงอ่านออกเสียงไปในทางอิตาเลียนว่า อาดูยโน่ หรือ บางคนก็อ่านว่า อาดูโน่ หรือ อาดูยอีโน่ ก็ได้ เริ่มต้นในปี 2005 ผู้ริเริ่มของ Arduino ชื่อว่า Massimo Banzi และ David Cuartielles ซึ่งอาศัยอยู่ในเมือง Ivrea ทางตะวันตกเฉียงเหนือของประเทศอิตาลี ทั้งสองคนตั้งใจสร้างอุปกรณ์ประเภทไมโครคอนโทรลเลอร์ราคาถูกที่นักเรียนนักศึกษาสามารถเข้าถึง และซื้อหามาเป็นเจ้าของได้ โรงงานเล็กๆ ในเมืองนี้ก็ถูกใช้เป็นที่ผลิตบอร์ด Arduino เวอร์ชั่นแรก โดยใช้ชื่อโครงการของพวกเขาว่า Arduin of Ivrea นอกจากจะตั้งใจให้ราคาของอุปกรณ์นั้นถูกเมื่อเทียบกับไมโครคอนโทรลเลอร์ตระกูลอื่นๆ ในท้องตลาดแล้วพวกเขายังตั้งใจให้ Arduino สามารถพัฒนาโดยโปรแกรมที่ "แจกฟรี" ภายใต้เงื่อนไขในการใช้งานลักษณะ Open Souce ดังนั้นจึงเลือกใช้การพัฒนาบนพื้นฐานของระบบ Wiringสำหรับบางคนที่ยังไม่แน่ใจว่าไมโครคอนโทรลเลอร์คืออะไร ไมโครคอนโทรลเลอร์ ว่าไปแล้วก็เปรียบเหมือนกับสมองของมนุษย์ คือมีหน้าที่คิด คำนวณทางคณิตศาสตร์ คำนวณทางลอจิกสั่งการมีส่วนความจำ เพื่อใช้เป็นข้อมูลในการคำนวณ หรือ ประมวลผลต่างๆ "แต่จะไม่สามารถทำงานได้เอง" โดยไม่มี มือ เท้า แขน ขา หรือ ตา หู จมูก ซึ่งเปรียบได้กับ อุปกรณ์ส่วนควบ (Accessories) อื่น เช่น เซนเซอร์ มอเตอร์ ระบบสื่อสารผ่านอินเตอร์เน็ตระบบแสดงผลผ่านจอภาพเป็นต้น ดังนั้นโดยสรุปคือ ไมโครคอนโทรลเลอร์จะทำหน้าที่ในการคิดคำนวณ รับค่าจากระบบวัดผลภายนอก เข้ามาประมวลผล เพื่อสั่งการตอบสนองออกไปที่อุปกรณ์ต่อเชื่อมอื่นๆ ไมโครคอนโทรเลอร์ในปัจจุบันก็มีอยู่หลายยี่ห้อ เช่น PIC ของบริษัทไมโครชิพ Z80 MCS-51 ARM-Cortex AVR และ อื่นๆอีกมากมาก Arduino ก็เป็นไมโครคอนโทรเลอร์แบบนึงที่มีเอกลักษณ์เฉพาะตัวที่ต่างจากยี่ห้ออื่นๆ คือการเป็น Open Source ซึ่งทำให้ได้เปรียบเรื่องราคา และจำนวนผู้ใช้งานทั่วโลก Arduino เวอร์ชั่นแรกปรากฏต่อสายตาชาวโลกในเดือนกันยายน ปี 2006 เรียกชื่อว่า Arduino Mini และก็เล็กแถมถูกสมชื่อ ราคาขายในเมืองไทยก็ตกอยู่ไม่กี่ร้อยบาท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถึงปัจจุบัน Arduinoมีบอร์ดหลายแบบให้เลือกใช้งานตามความถนัดและความเหมาะสม มากกว่า 20รุ่นแต่ละรุ่นก็มีขนาด ความจุ ความเร็ว จำนวนขาพอร์ตอินพุต เอาท์พุต แตกต่างกันออกไป มีตั้งแต่ราคาหลักสองสามร้อยบาท ไปจนกระทั่งพันกว่าบาทนอกจากนี้ยังมีอุปกรณ์ต่อพ่วง (Shield) ให้อีกสารพัด ราคาก็เป็นไปตาม concept คือ สมเหตุสมผลสุดๆ ทำให้เป็นที่นิยมมากขึ้นเรื่อยๆ



รูปที่ 2.20 แสดงตัวอย่างโปรแกรม Arduino

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.8.2 XAMPP

Xampp คืออะไร เป็นโปรแกรม Apache web server ไว้จำลอง web server เพื่อไว้ทดสอบสคริปหรือเว็บไซต์ในเครื่องของเราโดยที่ไม่ต้องเชื่อมต่ออินเตอร์เน็ตและไม่ต้องมีค่าใช้จ่ายใดๆ ง่ายต่อการติดตั้งและใช้งานโปรแกรม Xampp จะมาพร้อมกับ PHP ภาษาสำหรับพัฒนาเว็บแอพลิเคชั่นที่เป็นที่นิยม,MySQ ฐานข้อมูล,Apache จะทำหน้าที่เป็นเว็บเซิร์ฟเวอร์,Perl อีกทั้งยังมาพร้อมกับ OpenSSL ,phpMyadmin (ระบบบริหารฐานข้อมูลที่พัฒนาโดย PHP เพื่อใช้เชื่อมต่อไปยังฐานข้อมูลสนับสนุนฐานข้อมูล MySQL และ SQLite โปรแกรม Xampp จะอยู่ในรูปแบบของไฟล์ Zip,tar ,7z หรือexe โปรแกรม Xampp อยู่ภายใต้ใบอนุญาตของ GNU General Public License แต่บางครั้งอาจจะมีการเปลี่ยนแปลงเรื่องของลิขสิทธิ์ในการใช้งานจึงควรติดตามและตรวจสอบโปรแกรมด้วย

รูปที่ 2.21 สัญญาลักษณ์โปรแกรม XAMPP

โปรแกรม XAMPP สามารถใช้งานได้ 4 OS ได้แก่

2.8.2.1 Windows สามารถใช้งานได้กับ windows รุ่น 2000, 2003, xp, vista, windows 7 และจะมาพร้อมกับ

2.8.2.2 Linux สำหรับ SuSE, RedHat, Mandrake, Debian และ Ubuntu

2.8.2.3 Mac OS X

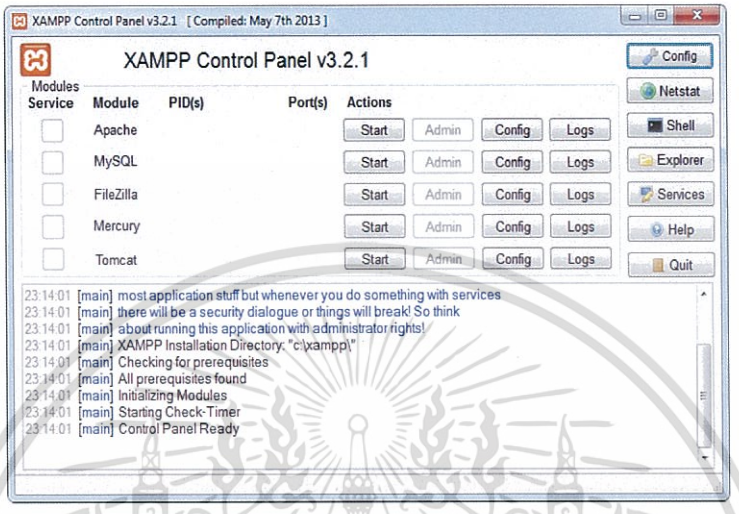
2.8.2.4 Solaris สำหรับ Solaris 8 และ Solaris 9

ในการใช้งานเบื้องต้นให้ดับเบิ้ลคลิ๊ก Xampp Control Panel Application แล้วทำการคลิ๊กปุ่ม start จากนั้นสามารถใช้งานได้โดยเปิดBrowser ขึ้นมาพิมพ์ localhost หรือ 127.0.0.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อจำกัดด้านเทคนิค

- เครื่องคอมพิวเตอร์ควรมี RAM ไม่ต่ำกว่า 128 MB
- Harddisk มีพื้นที่มากกว่า 320 MB
- CPU ไม่กำหนดขั้นต่ำ



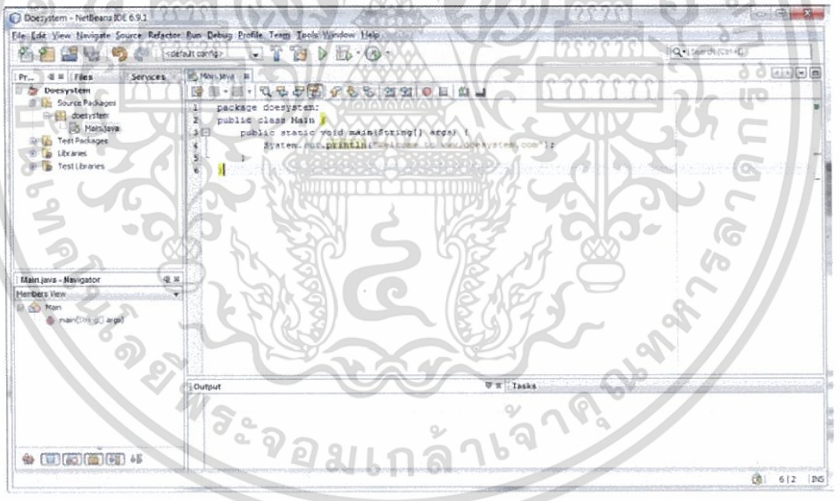
รูปที่ 2.22 ตัวอย่างโปรแกรม XAMPP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.8.3 Netbeans

NetBeans คือ เครื่องมือสำหรับโปรแกรมเมอร์ที่จะใช้พัฒนา Application ด้วยภาษา Java NetBeans นั้นเป็นโปรแกรมประเภท OpenSource software โดยผู้ใช้งานไม่จำเป็นที่จะต้องเสียเงินเพื่อซื้อมาใช้งานและยังเปิดเผย Sourcecodeให้ผู้สนใจและนักพัฒนานำไปดัดแปลงแก้ไขตามกฎของ Opensource โดยมี SunMicroSystem เป็นผู้สนับสนุนโครงการปัจจุบันNetBeansได้รับความนิยมมากยิ่งขึ้นและได้รับการพัฒนาให้มีความสามารถสูงยิ่งขึ้นเรื่อยๆจนถึงเวอร์ชั่นล่าสุด คือนอกจากจะใช้ในการพัฒนา Application ด้วยภาษาจาวาแล้ว ยังสามารถพัฒนาอื่นๆได้อีกหลากหลายโดยติดตั้งโปรแกรมเสริม(Add-on)ได้จากเว็บไซต์ หรือผ่านตัวอัปเดตเซนเตอร์ (Update Center) ของ NetBeans เช่น ภาษาซี/ซีพลัสพลัส(C/C++), Ruby, UML,SOA,Web Application, Java EE, Mobility(Java ME), Java FX, Java Script, PHP เป็นต้น ในเวอร์ชัน 6.0 เป็นต้นไปมีการรวมโปรแกรมเสริมต่างๆที่สำคัญเข้าในตัวติดตั้งของNetBeansโดยสามารถเลือกติดตั้งได้ภายหลัง

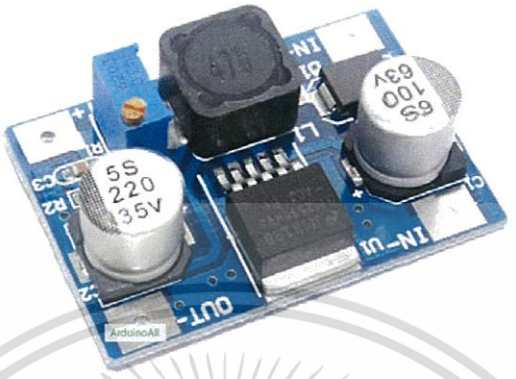
ข้อดีของโปรแกรมนี้ก็คือโปรแกรม NetBeans นั้นทำงานแยกส่วนต่างๆ ออกจากกันเป็น Module จึงทำให้สามารถนำ Module ต่างๆที่มีผู้ที่ได้พัฒนาต่อเติมมาติดตั้งเพิ่มเติมในภายหลังได้ ใช้งานได้กับระบบปฏิบัติการ Windows,Linux ,Mac OS X and Solaris



รูปที่ 2.23 ตัวอย่างโปรแกรม Netbeans

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.9 High voltage Step down convertor



รูปที่ 2.24 High voltage Step down convertor

LM2596HV High voltage Step down 5-60V to 1.25-26V

โมดูล LM2596HV แปลงไฟจาก 5-60 V (แนะนำที่น้อยกว่า 50V) ให้เป็น 1.25-26V กระแสสูงสุด 3A จ่ายกระแสต่อเนื่องที่ 1.2A พลังงานรวมแบบต่อเนื่องที่ 20W

ข้อมูล LM2596HV High voltage Step down 5-60V to 1.25-26V

- Input voltage range: 5V-60V (50V nominal recommend less reliable)
- Output voltage range: 1.25V-26V
- Maximum output current: 3A
- Continuous current: 1.2A (total output power less than 20W)
- Dimensions: L * W * H 4.3cm * 3.0cm * 1.2cm

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
การออกแบบ

3.1 การเก็บข้อมูลอุณหภูมิภายในและภายนอกของตู้ MSAN

การเก็บข้อมูลอุณหภูมิภายในและภายนอกของตู้ MSANนี้ เพื่อใช้ในการออกแบบชุดระบบควบคุมการระบายอากาศของตู้ MSAN ให้มีประสิทธิภาพและมีการประหยัดพลังงานอีกด้วย

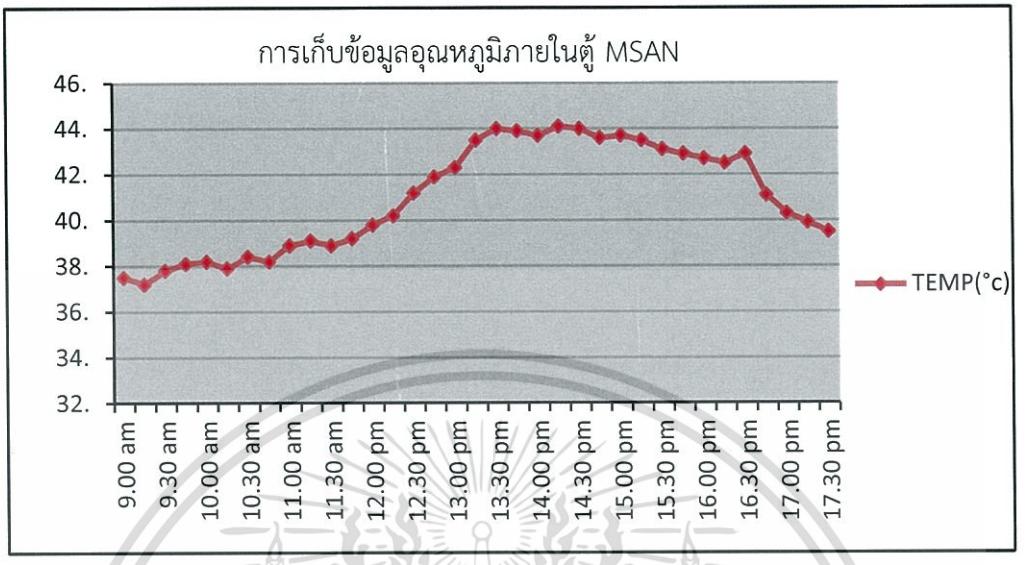
การวัดอุณหภูมิ ได้ใช้เทอร์โมมิเตอร์แบบ ดิจิตอลสามารถดึงข้อมูลผ่าน USB ได้และวัดอุณหภูมิได้ที่ ซอยสุคนธสวัสดิ์ 23 ถนนสุคนธสวัสดิ์ แขวงลาดพร้าว เขตลาดพร้าวกรุงเทพมหานคร 10230 และบรรยากาศโดยรอบตู้เป็นดังรูปด้วย

รูปที่ 3.1 ตู้ชุมสายโทรศัพท์ที่ใช้ทำการวัดอุณหภูมิ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.1 การเก็บข้อมูลอุณหภูมิภายในของตู้MSAN

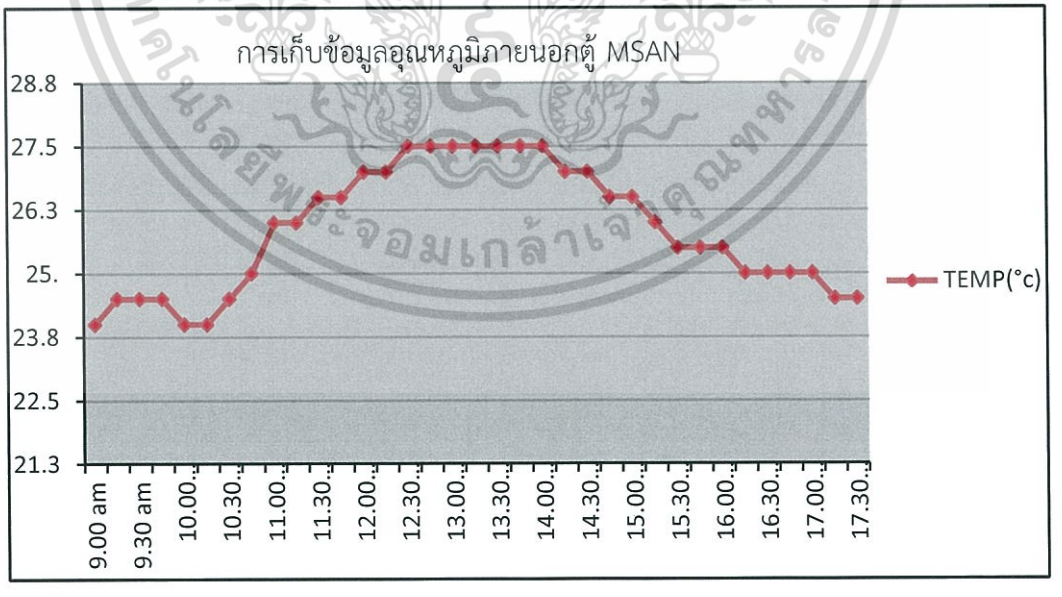
จากการวัดอุณหภูมิภายในตู้ MSAN ตั้งแต่เวลา 9.00 – 17.30 น. ได้ผลดังนี้



รูปที่ 3.2 อุณหภูมิภายในตู้ MSAN

3.1.2 การเก็บข้อมูลอุณหภูมิภายนอกของตู้MSAN

จากการวัดอุณหภูมิภายนอกตู้ MSAN ตั้งแต่เวลา 9.00 – 17.30 น. ได้ผลดังนี้

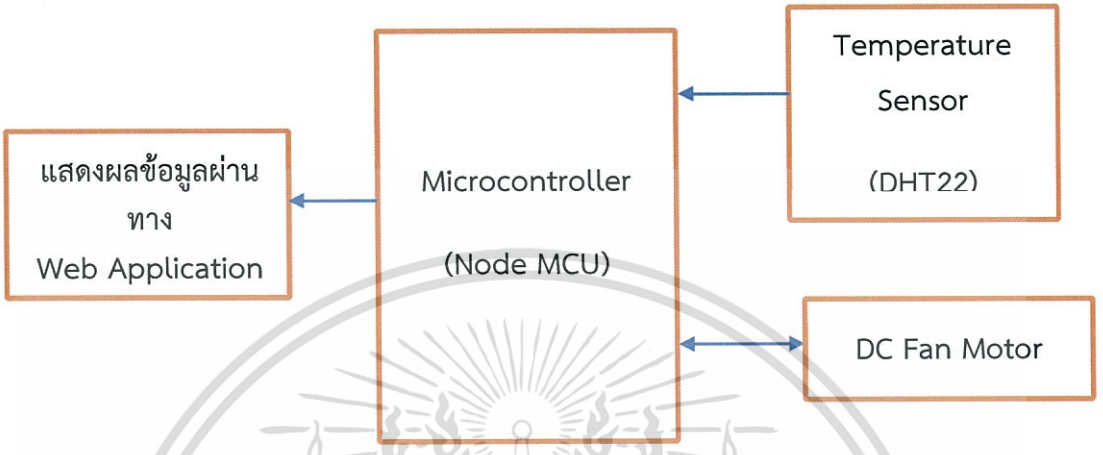


รูปที่ 3.3 อุณหภูมิภายนอกตู้ MSAN

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 โครสร้างการทำงานของระบบระบายอากาศ

การพัฒนาระบบระบายอากาศนั้นถูกแบ่งการทำงานออกเป็น 3 ส่วน ได้แก่ ส่วนควบคุมความเร็วพัดลมตามอุณหภูมิ, ส่วนวัดความเร็วรอบพัดลม และส่วนแสดงผลข้อมูลทาง Web Application โดยจะใช้ไมโครคอนโทรลเลอร์(Node MCU) เป็นตัวควบคุมการทำงานแต่ละส่วน

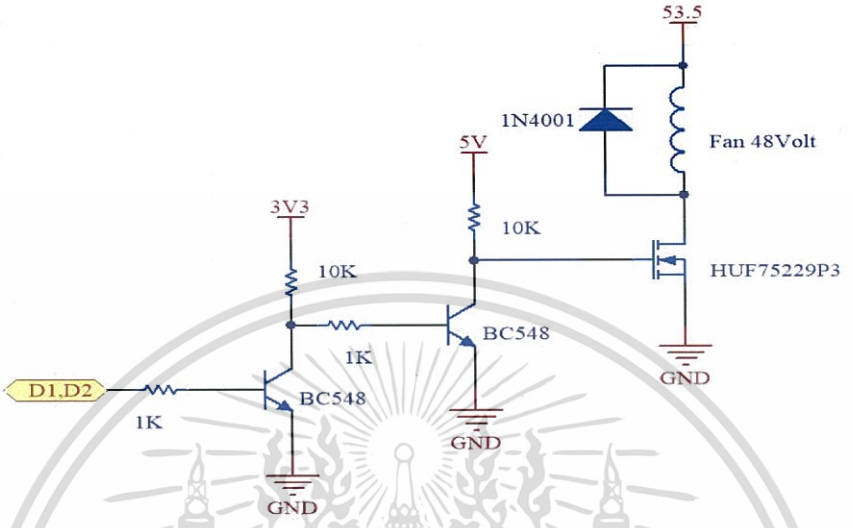


รูปที่ 3.4 โครงสร้างการทำงานของระบบระบายอากาศ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 การออกแบบส่วนควบคุมความเร็วพัดลมตามอุณหภูมิ

3.3.1 ส่วนวงจรควบคุมความเร็วพัดลมตามอุณหภูมิ



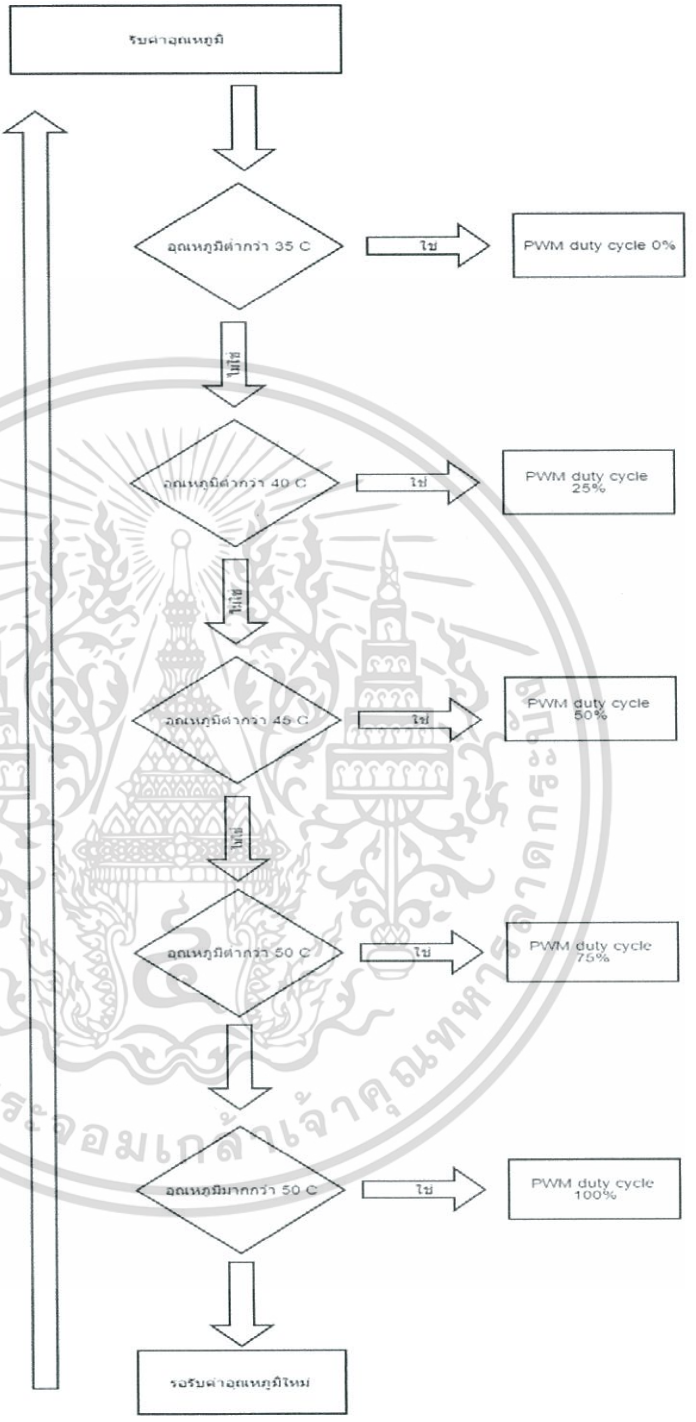
รูปที่ 3.5 วงจรควบคุมความเร็วพัดลมตามอุณหภูมิ

ในการออกแบบนี้เราใช้เซนเซอร์ DHT22 ในการวัดอุณหภูมิ ตัวเซนเซอร์นี้จะทำการส่งข้อมูลที่ทำการวัดอุณหภูมิเข้ากับไมโครคอนโทรลเลอร์ Node MCU ESP8266 โดยจะมีการโปรแกรมในตัวไมโครคอนโทรลเลอร์นี้ให้ส่งสัญญาณ PWM และ Duty cycle ไปควบคุมพัดลมมอเตอร์กระแสตรงให้ปรับความเร็วตามอุณหภูมิได้

ไมโครคอนโทรลเลอร์ Node MCU ESP8266 ทำการส่งสัญญาณ PWM ขนาด 3.3 Vpp เข้าสู่ขาเบสที่ทรานซิสเตอร์เพื่อทำการขยายสัญญาณ PWM เป็น 5 Vpp และนำไปขับมอสเฟตที่ขาเกตให้มอสเฟตเกิดการทำงานเพื่อที่จะไปควบคุมพัดลมมอเตอร์กระแสตรง ส่วนที่ขาซอร์สจะเชื่อมต่อขากราวด์ และขาเดรนจะเชื่อมต่อกับขากราวน์ดของพัดลมมอเตอร์กระแสตรง และส่วนขาไฟเลี้ยงพัดลมมอเตอร์กระแสตรงจะเชื่อมเข้ากับแหล่งจ่ายไฟ 48VDC โดยมี Free wheel diode คร่อมพัดลมมอเตอร์กระแสตรงเพื่อป้องกันการเกิดกระแสไหลย้อนกลับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.2 ผังงานโปรแกรมควบคุมความเร็วพัดลมตามอุณหภูมิ



รูปที่ 3.6 ผังงานควบคุมความเร็วพัดลมตามอุณหภูมิ

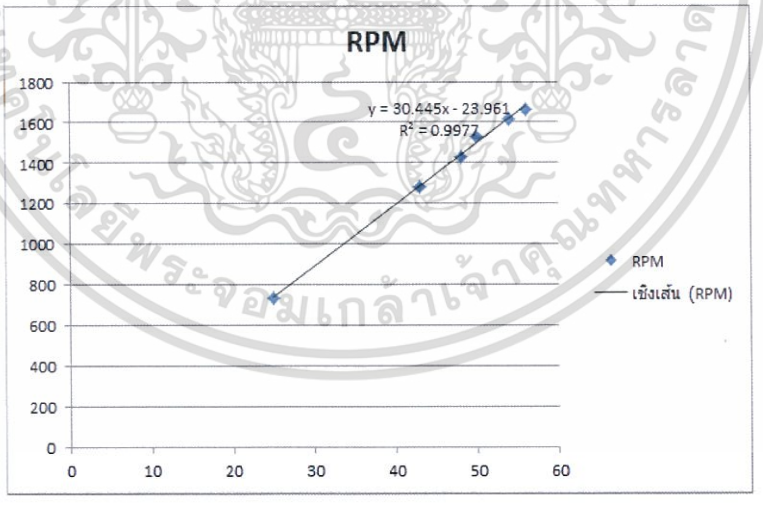
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 การออกแบบส่วนวัดความเร็วรอบพัดลม

พัดลมที่ใช้ในระบบระบายอากาศ จะมีเซนเซอร์ตรวจจับการหยุดของพัดลม เซนเซอร์ตัวนี้สามารถนำมาประยุกต์ใช้ในการวัดความเร็วรอบของพัดลมได้โดยทำการจ่ายพลังงานที่แตกต่างกันและทำการวัดความถี่ของสัญญาณจากเซนเซอร์ของพัดลมและใช้เครื่องมือในการวัดรอบพัดลมในการวัด ค่าความเร็วรอบของพัดลม นำข้อมูลจากข้างต้นนั่นมาสร้างกราฟความสัมพันธ์ระหว่างความถี่และความเร็วรอบ

volt	frequency	RPM
19	25	733
20	43	1285
21	48	1431
22	50	1529
23	54	1619
24	56	1662

ตารางที่ 3.1 ข้อมูล volt ต่อ frequency และค่า RPM



รูปที่ 3.7 ความสัมพันธ์ระหว่างความถี่กับความเร็วรอบพัดลม

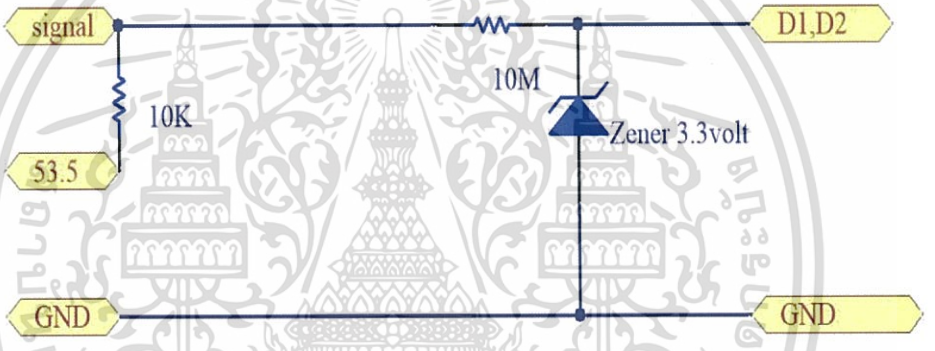
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากการสร้างกราฟความสัมพันธ์ระหว่างความถี่กับความเร็วรอบพัดลม จะสังเกตุได้ว่าเส้นกราฟมีความใกล้เคียงสมการเส้นตรง ทำให้สร้างสมการในการหาความเร็วรอบจากความถี่ได้ดังนี้

$$RPM = 30.445(Frequency) - 23.961$$

แต่ภายในเซนเซอร์สัญญาณที่ออกมาจะไม่มีกระแสไหล ทำให้ต้องใช้ วิธีการ Pull-up ในการทำให้มีกระแสในเส้นสัญญาณ โดยทำการต่อความต้านทานระหว่างสัญญาณกับเส้นพลังงงานของพัดลมหลังจากนั้นใช้ตัวต้านทานอีกตัวหนึ่งและ ซีเนอร์ไดโอดเพื่อป้องกัน NodeMCU ได้รับความเสียหาย

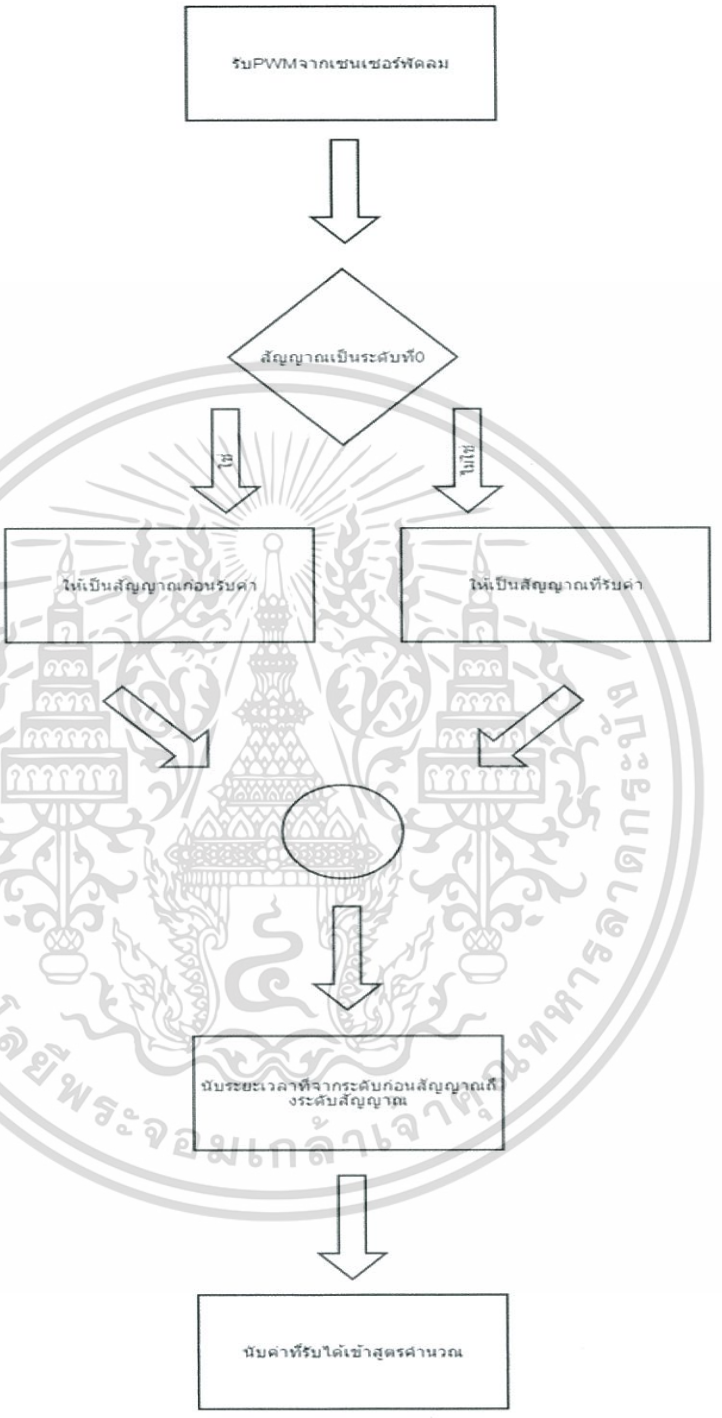
NodeMCU จะทำการวัดความถี่ จากสัญญาณ เพื่อที่จะให้NodeMCU จะคำนวณความเร็วรอบของพัดลม จากความถี่ของสัญญาณที่เข้าสู่สูตรคำนวณข้างต้น



รูปที่ 3.8 วงจรวัดความเร็วรอบพัดลมจากสัญญาณพัดลมของพัดลม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4.1 ผังงานโปรแกรมส่วนวัดความเร็วรอบพัดลม



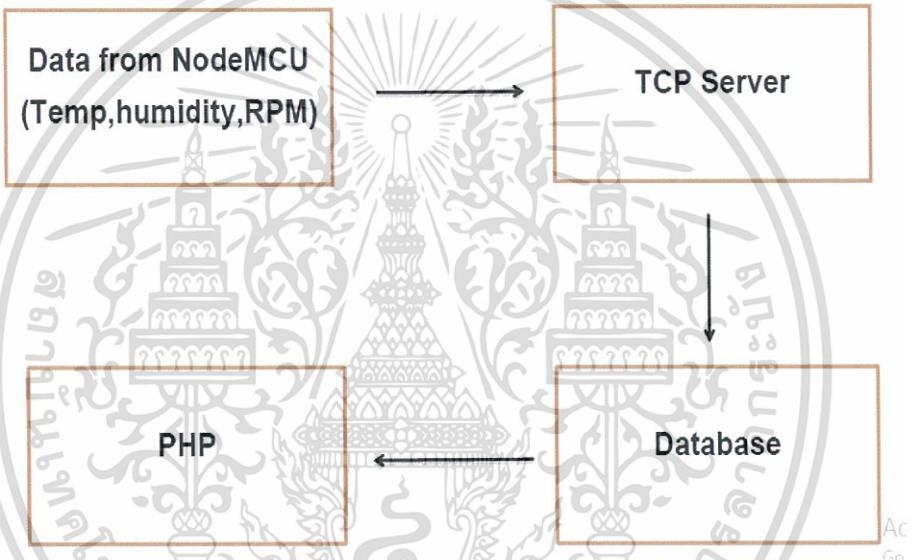
รูปที่ 3.9 ผังงานการวัดความเร็วรอบของพัดลม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5 การออกแบบส่วนแสดงข้อมูลบน Web Application

เนื่องจากไมโครคอนโทรลเลอร์ Node MCU ESP8266 มีโมดูล Wi-Fi ในตัวจึงทำให้ไมโครคอนโทรลเลอร์สามารถเชื่อมต่อกับระบบ Wi-Fi ได้อย่างง่ายดายผ่าน AT Command สามารถรับค่าจาก Sensor ต่างๆ ประมวลผล และรับส่งข้อมูลได้ด้วยตัวมันเพียงลำพัง

ในที่นี้ไมโครคอนโทรลเลอร์จะทำการเชื่อมเข้ากับตัว TCP Server ซึ่งตัว server จะรับค่าที่วัดได้จากไมโครคอนโทรลเลอร์คือ Temperature, Humidity, RPMI และ RPMII จากนั้น server จะนำค่าที่ได้มาเก็บข้อมูลทั้งหมดลง Database และนำข้อมูลที่เก็บมาจาก database แสดงผลบน Web Application

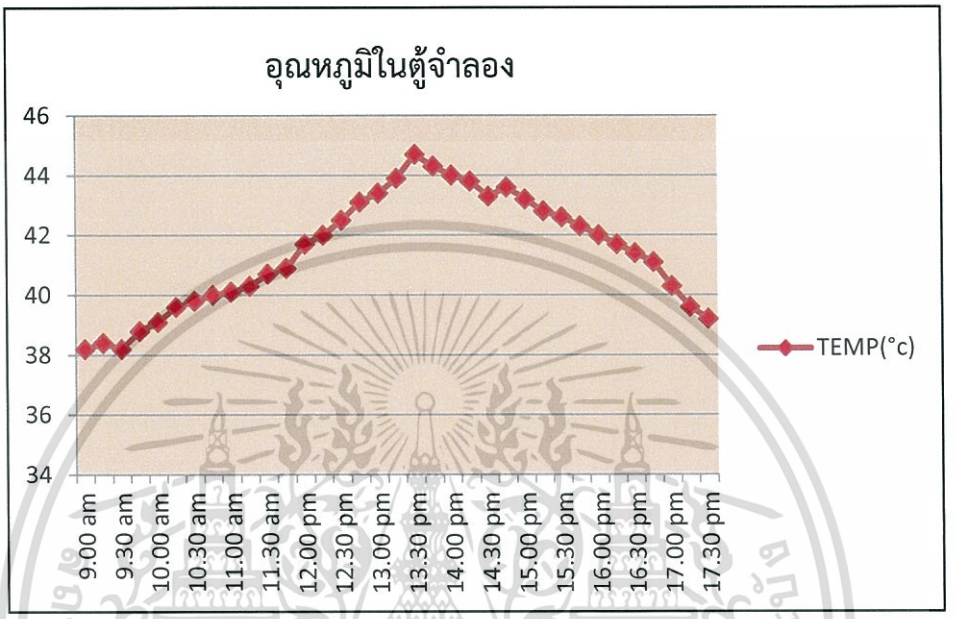


รูปที่ 3.10 Block diagram sent data on Web Application

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
ผลการทดลอง

4.1 กราฟแสดงอุณหภูมิจากการจำลองตู้ MSAN



รูปที่ 4.1 ความสัมพันธ์ระหว่างอุณหภูมิกับเวลาในตู้จำลอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

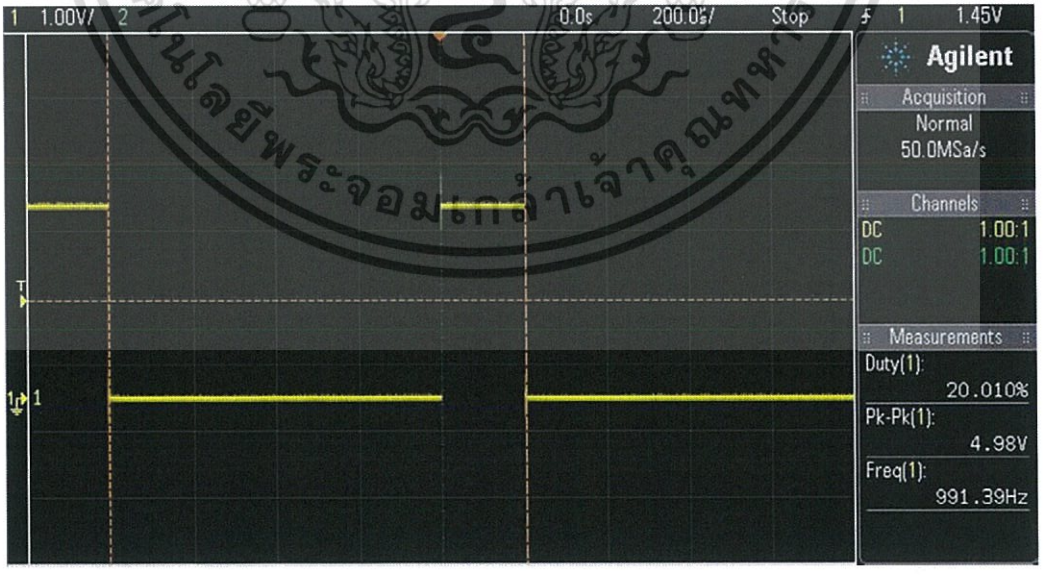
4.1.1 ค่าอุณหภูมิและเวลาที่นำไปพอทลงบนความสัมพันธ์ระหว่างอุณหภูมิกับเวลา

TIME	TEMP	TIME	TEMP	TIME	TEMP
9.00 am	38.2	12.00 pm	41.7	15.00 pm	43.2
9.15 am	38.4	12.15 pm	42	15.15 pm	42.8
9.30 am	38.2	12.30 pm	42.5	15.30 pm	42.6
9.45 am	38.8	12.45 pm	43.1	15.45 pm	42.3
10.00 am	39.1	13.00 pm	43.4	16.00 pm	42
10.15 am	39.6	13.15 pm	43.9	16.15 pm	41.7
10.30 am	39.8	13.30 pm	44.7	16.30 pm	41.4
10.45 am	40	13.45 pm	44.3	16.45 pm	41.1
11.00 am	40.1	14.00 pm	44	17.00 pm	40.3
11.15 am	40.3	14.15 pm	43.8	17.15 pm	39.6
11.30 am	40.7	14.30 pm	43.3	17.30 pm	39.2
11.45 am	40.9	14.45 pm	43.6		

ตารางที่ 4.1 ค่าอุณหภูมิและเวลาที่นำไปพอทลงบนความสัมพันธ์ระหว่างอุณหภูมิกับเวลา

4.2 ผลการทดลองส่วนควบคุมพัดลมตามอุณหภูมิ

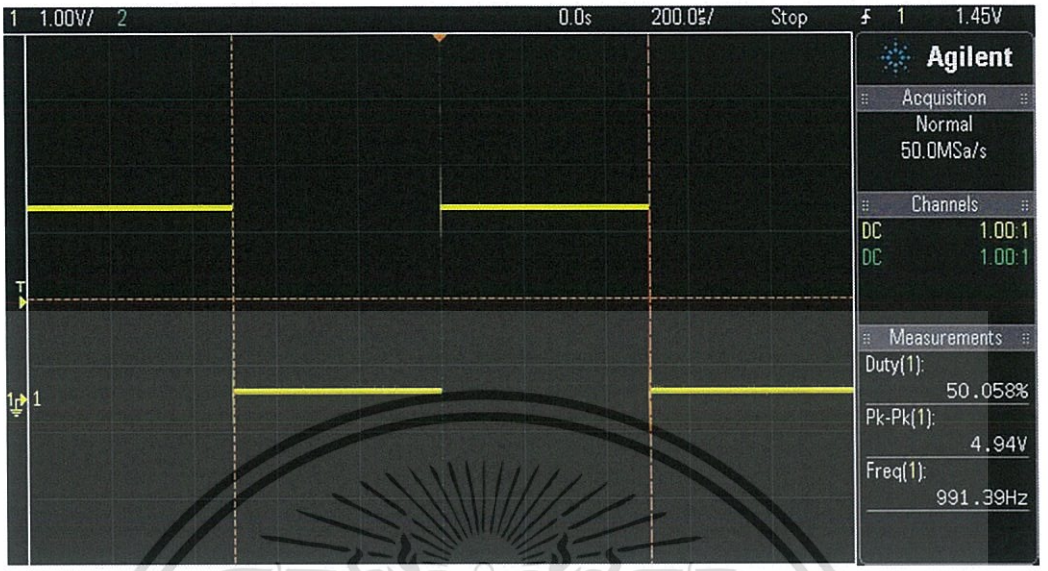
4.2.1 ที่อุณหภูมิ < 40c duty cycle ที่วัดได้เท่ากับ 20%



รูปที่ 4.2 duty cycle ที่วัดได้เท่ากับ 20%

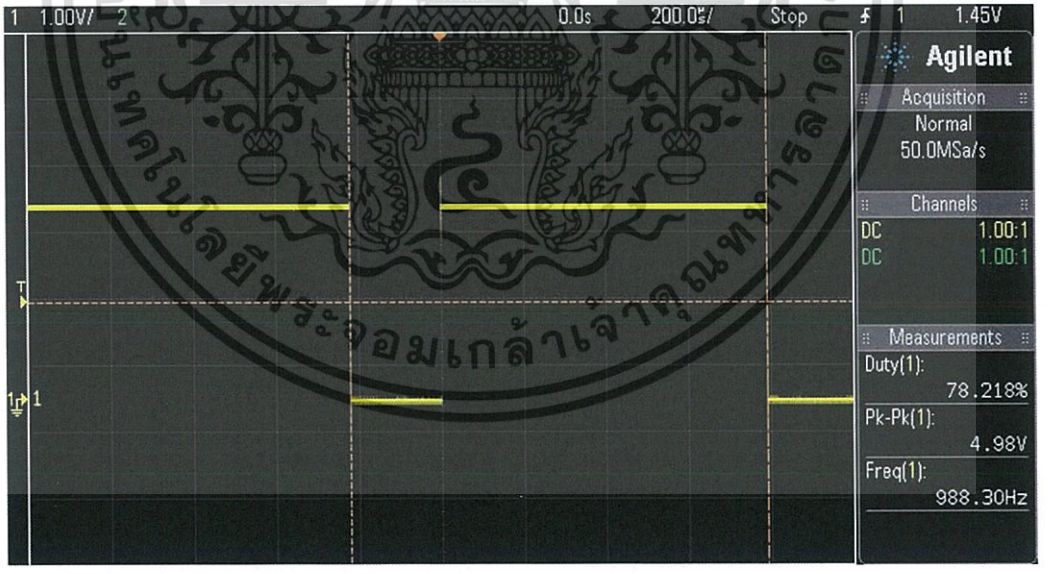
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.2 ที่อุณหภูมิ < 45c duty cycle ที่วัดได้เท่ากับ 50%



รูปที่ 4.3 duty cycle ที่วัดได้เท่ากับ 50%

4.2.3 ที่อุณหภูมิ < 50c duty cycle ที่วัดได้เท่ากับ 80%



รูปที่ 4.4 duty cycle ที่วัดได้เท่ากับ 80%

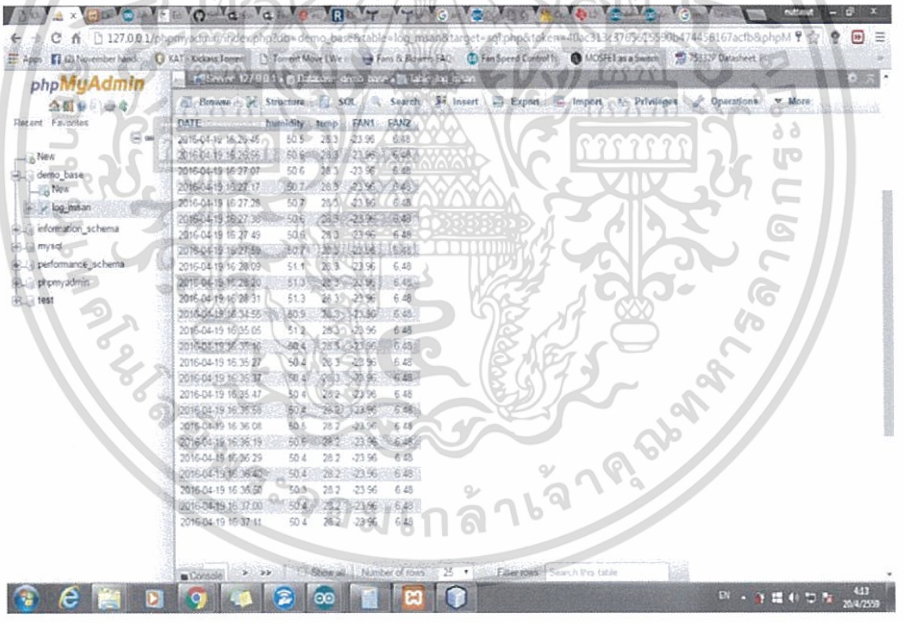
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 ผลการทดลองส่วนวัดความเร็วรอบพัดลม

Duty cycle	ค่าที่วัดได้จาก ไมโครคอนโทรลเลอร์	ค่าที่วัดได้จากเครื่องมือวัด
20%	1316	1251
50%	1591	1523
80%	2428	2346
100%	3012	2934

ตารางที่ 4.2 ผลการทดลองส่วนวัดความเร็วรอบพัดลม

4.4 ข้อมูลจาก Database และค่าที่แสดงผลทาง Web Application

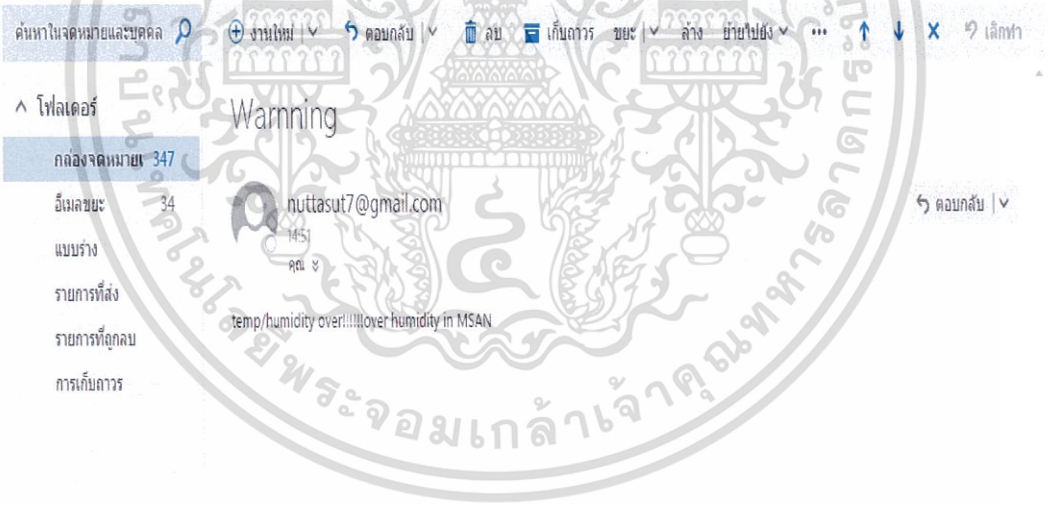


รูปที่ 4.5 ข้อมูลจาก Database

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.6 ค่าที่แสดงผลทาง Web Application

4.5 ผลของการส่งข้อมูลผ่าน E-mail



รูปที่ 4.7 ข้อมูลที่แจ้งเตือนมาทาง E-mail

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
บทสรุป

ในปริญญานิพนธ์นี้ได้กล่าวถึงการพัฒนาระบบระบายความร้อนในตู้ MSAN จากการศึกษาและการออกแบบระบบระบายอากาศในตู้ MSAN ในด้านการควบคุมความเร็วของพัดลม ตามอุณหภูมิที่ได้จากการทดลองเราจะสามารถปรับความเร็วของพัดลมในขณะที่อุณหภูมิภายในตู้ MSAN มีอุณหภูมิสูง โดยพัดลมจะมีความเร็วในการหมุนเพิ่มขึ้น โดยการปรับ duty cycle ให้สูงขึ้นตามอุณหภูมิที่ได้โปรแกรมไว้ ในไมโครคอนโทรเลอร์ สำหรับการวัดความเร็วรอบของพัดลม จากการทดลองเราสามารถวัดความเร็วที่ได้จากสัญญาณที่ออกมาจากตัวพัดลมนั้น จะมีความใกล้เคียงกับค่าความเร็วรอบพัดลมที่วัดได้จากเครื่องวัดความเร็วรอบ และสำหรับด้านการแสดงผลบน Web Application เราสามารถนำข้อมูลทั้ง อุณหภูมิ ความชื้น และความเร็วรอบของพัดลม ที่ได้จากตัวไมโครคอนโทรลเลอร์นั้น มาเก็บข้อมูลและสามารถแสดงข้อมูลดังกล่าวภายใน Web Application ได้

โครงการนี้จะได้ผลดียิ่งขึ้น โดยทำการทดลองกับตู้ชุมสายโทรศัพท์จริง เนื่องจากการทำโครงการนี้ต้องมีการทดสอบเพื่อให้ได้ผลและคุณภาพที่ดียิ่งๆขึ้นไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

[1] รศ.ดร.ชูชาติ ปิณฑวิรุจน์, Signals and Systems, พิมพ์ครั้งที่ 3.
มีน เซอร์วิส ซัพพลาย. 2556
[2] K.Alexander & Matthew N.O.Sadiku, Fundamentals of Electric
Circuits by The McGraw-hill Companiesม 2007
[3] วรากร เกษมสุวรรณ์, Analysis of Analog CMOS integrated Circuits,
พิมพ์ครั้งที่ 2. กรุงเทพฯ : สาขาวิชาอิเล็กทรอนิกส์ คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง, 2553
[4] ประภากร สุวรรณะ, Electronic Engineering I, กรุงเทพฯ.สาขาวิชา อิเล็กทรอนิกส์ คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง,2545
[5] Owen Bishop,Electronic-Circuits and systems,Third Edition.,
Newnes Newton,MA,USA 2007
[6] ประจิน พลังสันติกุล, C Programming for AVR microntroller,
พิมพ์คที่1. กรุงเทพ : แอพซอฟต์แทค, 2549
[7] M.H. Rashid, Power Electronics, devices and Application,Third Editon.,New
Jersey,Pearson Education, 2004
[8] **High voltage Step down** ค้นเมื่อ 20 มกราคม 2016, จาก
http://www.arduinoall.com/product/855/lm2596hv-high-voltage-step-down-5-60v-to-1-25-26v
[9] **NodeMCU** ค้นเมื่อ 24 มกราคม 2016, จาก http://www.thaieasyelec.com/article-wiki/embedded-electronics-application/getting-started-with-esp8266-nodemcu.html
[10] **HUF75229P3** ค้นเมื่อ 18 มกราคม 2016,
จาก HTTPS://WWW.FAIRCHILDSEMI.COM/DATASHEETS/HU/HUF75229P3.PDF
[11] **BC548** 18 มกราคม 2016, จาก HTTP://WWW.ALLDATASHEET.COM/DATASHEET-PDF/PDF/11552/ONSEMI/BC548.HTML

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. ส่วนโปรแกรมควบคุมความเร็วพัดลมตามอุณหภูมิ

```
#include <DHT.h>
#define DHTPIN D5
#define DHTTYPE DHT22
DHT dht(DHTPIN, DHTTYPE);
int mortorI = D7;
int mortorII = D4;
void setup()
{
pinMode(mortorI, OUTPUT);
        pinMode(mortorII, OUTPUT);
}
void loop()
{
delay(2000);
float hum = dht.readHumidity();
float temp= dht.readTemperature();
if(temp <35)
 {
  analogWrite(mortorI, 0);
  analogWrite(mortorII, 0);
 }
 else if (temp<40)
 {
  analogWrite(mortorI, 256);
  analogWrite(mortorII, 256);
 }
  else if (temp<45)
 {
    analogWrite(mortorI, 512);
    analogWrite(mortorII, 512);
   }
   else if(temp<50)
   {
    analogWrite(mortorI, 768);
    analogWrite(mortorII, 768);
   }
   else if (temp>50)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  {
   analogWrite(mortorI, 1024);
   analogWrite(mortorII, 1024);
  }
}
```


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. ส่วนโปรแกรมควบคุมความเร็วพัดลมตามอุณหภูมิ

```
const int frqI = D1;
const int frqII = D2;
const int oneSecond = 1000;
int frqIState = LOW;
int frqIIState = LOW;
int frq1 = 0;
int frq2 = 0;
int prevFrqIState = LOW;
int prevFrqIIState = LOW;
unsigned long timer = 0;
void setup()
{
pinMode(frqI, INPUT);
pinMode(frqII, INPUT);
}
frqIState = digitalRead(frqI);
 if (frqIState == LOW)
 {
   prevFrqIState = LOW;
 }
 if (frqIState == HIGH && prevFrqIState == LOW)
 {
   prevFrqIState = frqIState;
   frq1++;
 }
 if (millis() - timer > oneSecond)
 {
   timer = millis();
   frq1 = 0;
 }
 float rpmI = (30.445*(frq1) - 23.961);
 frqIIState = digitalRead(frqII);
 if (frqIIState == LOW)
 {
   prevFrqIIState = LOW;
 }
 if (frqIIState == HIGH && prevFrqIIState == LOW)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{
  prevFrqIIState = frqIIState;
  frq2++;
}
if (millis() - timer > oneSecond)
{
  timer = millis();
  frq2 = 0;
}
float rpmII = (30.445*(frq2) - 23.961);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1 การโปรแกรมในไมโครคอนโทรลเลอร์เพื่อเชื่อมต่อกับ TCP server

```
#include <ESP8266WiFi.h>
const char* ssid = "Connectify-me";
const char* password = "tajifuj1";
#define SERVER_PORT 12349
IPAddress server_ip = {192,168,22,1};
WiFiServer server(SERVER_PORT);
WiFiClient c void setup()
void setup()
{
    WiFi.begin(ssid, password);
   while (WiFi.status() != WL_CONNECTED)
    {
            delay(500);
            Serial.print(".");
    }
    Serial.println("");
    Serial.println("WiFi connected");
    Serial.println("IP address: ");
    Serial.println(WiFi.localIP());

    Serial.println("Connect TCP Server");
    while (!client.connect(server_ip,SERVER_PORT))
    {
            Serial.print(".");
            Serial.print(SERVER_PORT);
            delay(100);
    }
    Serial.println("Success");
 ESP.wdtDisable();
 void loop()
 {
 client.print("Humidity: ");
  delay(1000);
            Serial.println(hum);
            String HUM = String(hum);
 delay(1000);
            client.print(" %, Temparature: ");
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        delay(1000);
        Serial.println(temp);
        String TEMP = String(temp);
        delay(1000);
        client.println(" C");
        delay(1000);
        Serial.println("rpm_fan1:");
        delay(1000);
        Serial.println(rpmI);
        String RPMI = String(rpmI);
        delay(1000);
        client.print(" rpm_fan2:");
        delay(1000);
        Serial.println(rpmII);
         String RPMII = String(rpmII);
        HUM +=':';
        HUM +=TEMP;
        HUM +=':';
        HUM +=RPMI;
        HUM +=':';
        HUM +=RPMII;
        client.print(HUM);
        Serial.println(HUM);

        delay(60000);
}
    c, addr = s.accept()     # Establish connection with client.
    print 'Got connection from', addr
    data = c.recv(1024)
    print data + "Main loop data"
    SQL(data)
    #fuction()
    # Close the connection
finally:
    c.close()
    s.close()
    print 'End'
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 การโปรแกรมการเชื่อมต่อ TCP server กับ การเก็บข้อมูลใน database

```
import socket
import time
import MySQLdb
import datetime
def Split_data(raw_dat)
split_data = raw_data.split(":")
    #print split_data
    return split_data
def get_date():
    now = datetime.datetime.now()
    date = str("%d-%d-%d"%(now.year,now.month,now.day))
    Time = str("%d:%d:%d"%(now.hour,now.minute,now.second))
    return date+" "+Time

def SQL(data):
    split_data = Split_data(data)
    date_time = get_date()
    print "split_data"
    print split_data
    print "date_time"
    print date_time
    db = MySQLdb.connect("localhost","root","1234","demo_base" )
    print "Connected to SQL"
          cursor = db.cursor()
 sql = """INSERT INTO log_Msan(DATE,humidity,temp,FAN1,FAN2)
          VALUES('%s','%s','%s','%s','%s')"""%
(date_time,split_data[0],split_data[1],split_data[2],split_data[3])#printsql
    try: # Execute the SQL command
       cursor.execute(sql)
       db.commit()
       print "Update completed"
    except:
       db.rollback()
 db.close()
 ##########################################################################
 s = socket.socket()
 host = '192.168.22.1'
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
port = 12349
s.bind((host, port))
s.listen(5)
try: # Now wait for client connection.
        while True:
        print 'Wait connection'
        c, addr = s.accept()     # Establish connection with client.
        print 'Got connection from', addr
        data = c.recv(1024)
        print data + "Main loop data"

 SQL(data)
 #fuction()
     # Close the connection
finally:
    c.close()
    s.close()
print 'End'
            time.sleep(10)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 การโปรแกรมการนำข้อมูลใน database แสดงผลบนWebApplication

```
<html>
  <head>
    <title>Last 10 Results</title>
  </head>
  <body>
    <h1 align="center" >MSAN data log</h1>
    <table border="1" style="width:60%" align="center">
        <tr>
          <th>DATE</th>
          <th>Humidity</th>
          <th>Temp</th>
          <th>FAN1</th>
          <th>FAN2</th>
        </tr> <?php
      $servername = "localhost";
      $username = "root";
      $password = "1234";
      $dbname = "demo_base";

      // Create connection
      $conn = new mysqli($servername, $username, $password, $dbname);
      // Check connection
      if ($conn->connect_error) {
          die("Connection failed: " . $conn->connect_error);
      }
$sql = "SELECT * FROM log_Msan";
      $sql = "SELECT * FROM log_Msan ORDER BY DATE ASC LIMIT 50";
      $result = $conn->query($sql);

      while($row = $result->fetch_assoc())
      {
          <tr>
              <tr align="center">
              <td><?php echo $row['DATE']?></td>
              <td><?php echo $row['humidity']?></td>
              <td><?php echo $row['temp']?></td>
              <td><?php echo $row['FAN1']?></td>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                <td><?php echo $row['FAN2']?></td>
            </tr>
            </tr>
<?php
        }
        ?>
</table>
    </body>
</html>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้