พัดลมปรับความเร็วตามอุณหภูมิ

Adjustable Speed Fan by Temperature

จัดทำโดย

นรจ.วัชรพล บุญพินิจ
นรจ.ภานุพงศ์ ไชยสลี
นรจ.ภานุเดช อาบทอง
นรจ.รัฐศาสตร์ กัลโยธิน

โครงงานนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรนักเรียนจ่า
พรรคพิเศษ เหล่าทหารช่างยุทธโยธา (อิเล็กทรอนิกส์)
โรงเรียนอิเล็กทรอนิกส์ กองวิทยาการ กรมอิเล็กทรอนิกส์ทหารเรือ
ปีการศึกษา ๒๕๖๕

หัวข้อโครงงาน พัดลมปรับความเร็วตามอุณหภูมิ
Fan speed adjusts according to temperature

ผู้จัดทำ นรจ.วัชรพล บุญพินิจ
นรจ.ภานุพงศ์ ไชยสลี
นรจ.ภานุเดช อาบทอง
นรจ.รัฐศาสตร์ กัลโยธิน

ครูที่ปรึกษา ว่าที่ น.อ.วิชัย เปลี่ยนสุวรรณ
ว่าที่ ร.ต.ไกรสรณ์ รื่นเหลย
พ.จ.อ.ชินเรษ วิสียา

ปีการศึกษา ๒๕๖๕

บทคัดย่อ

โครงงานนี้เป็นโครงงานพัดลมปรับความเร็วตามอุณหภูมิ เพื่อนำความรู้ภาคทฤษฎีมาบูรณาการใช้ในการทำโครงงาน โดยใช้หลักการ Sensor ตรวจจับค่าอุณหภูมิและค่าความชื้นสัมพัทธ์แล้วประมวลผลเพื่อทำการปรับระดับความเร็วของพัดของพัดลม มีส่วนประกอบหลักคือ Relay Board, Power supply 5V, พัดลมตั้งพื้น, Node MCU ESP8266 และ DHT 22

การทำงานมี 2 ระบบ คือระบบ Manual และระบบ Automatic โดยระบบ Manual จะทำการปรับระดับความเร็วของพัดลมผ่านทาง BLYNK Application เมื่อ BLYNK App มีปัญหาหรือสัญญาณ wireless ไม่เสถียร ก็จะเปลี่ยนไปใช้การทำงานแบบ Manual ได้ โดยการกดสวิตช์พัดลมได้ตามปกติค่าความชื้นที่บันทึกไว้ใน BLYNK App เมื่อเซ็นเซอร์ทำการตรวจวัดค่าความชื้น ค่าที่ได้จะนำไปประมวลผลเพื่อทำการปรับความเร็วใบพัดของพัดลมจากทดลองถ้าค่าอุณหภูมิมีค่าต่ำพัดลมจะหมุนช้าลงและค่าอุณหภูมิสูง ใบพัดลมจะหมุนเร็วขึ้น

.....
ครูที่ปรึกษาโครงงาน

กิตติกรรมประกาศ

การจัดทำโครงงานสิ่งประดิษฐ์ในครั้งนี้สำเร็จได้โดยการค้นคว้าหาความรู้ การให้คำปรึกษาและการกำกับดูแลจาก ว่าที่ น.อ.วิชัย เปลี่ยนสุวรรณ ว่าที่ ร.ต.ไกรสรณ์ รื่นเหลย และ พ.จ.อ.ชินเรษ วิสียา ครูที่ปรึกษาโครงงาน ขอขอบพระคุณคณะครูฝ่ายศึกษาที่ให้การสนับสนุนในด้านความรู้และคำแนะนำ รวมถึงสนับสนุนอุปกรณ์ที่ขาดเหลือเกี่ยวกับโครงงานสิ่งประดิษฐ์ ขอขอบพระคุณคณะครูห้องวิทยาการที่ให้การสนับสนุนและคำแนะนำเรื่องการเขียนโปรแกรม และควบคุมอุปกรณ์ที่ใช้ในการทำโครงงงานสิ่งประดิษฐ์ คุณความดีที่เกิดจากโครงงานสิ่งประดิษฐ์นี้ ขอมอบให้กับคณะครูและอาจารย์ทุกท่านที่ให้การสนับสนุนและประสิทธิ์ประสานวิชาทำให้นักเรียนจ่ามีความรู้ ความเข้าใจส่งผลให้การทำโครงงานสิ่งประดิษฐ์ชิ้นนี้สำเร็จลุล่วงไปได้ด้วยดี

จัดทำโดย

นรจ.วัชรพล บุญพินิจ

นรจ.ภานุพงศ์ ไชยสลี

นรจ.ภานุเดช อาบทอง

นรจ.รัฐศาสตร์ กัลโยธิน

สารบัญ

บทคัดย่อ ก

กิตติกรรมประกาศ ข

บทที่ 1 บทนำ 1

1.1 ที่มาและความสำคัญ 1

1.2 วัตถุประสงค์ 1

1.3 ขอบเขตการทำโครงงาน 1

1.4 ประโยชน์ที่ได้รับจากโครงงาน 1

บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวของ 2

2.1 เซ็นเซอร์วัดอุณหภูมิและความชื้น DHT22 2

2.2 รีเลยโมดูล (Relay Module) 4 ช่อง 5 โวลต์ 10A 250V 3

2.3 บอรดไมโครคอนโทรลเลอร์ ESP8266 Node MCU V3 4

2.4 จอ LCD 6

2.5 Power supply 5v 6

บทที่ 3 วิธีการดำเนินงาน 7

3.1 วิธีการดำเนินงาน 8

3.2 แผนการดำเนินงาน 11

3.3 วัสดุอุปกรณ์ 12

บทที่ 4 ผลการทดลอง 13

4.1 ตารางแบบสอบถามความต้องการระดับพัดลม 13

4.2 การทดลองพัดลมปรับความเร็วตามอุณหภูมิและผ่าน Blynk 14

สรุปผลการทดลอง 14

บทที่ 5 ปัญหาและข้อเสนอแนะ 15

5.1 สรุปโคงงาน 15

5.2 ปัญหา 15

5.3 ข้อเสนอแนะ 15

บรรณานุกรม 16

ภาคผนวก 17

สารบัญรูป

เรื่อง **หน้า**

รูปที่ 2.1 เซ็นเซอร์ DHT 22 2
รูปที่ 2.2 รีเลย์โมดูล 3
รูปที่ 2.3 แสดงสภาวะการทำงานของรีเลย์ 4
รูปที่ 2.4 Node MCU Version3 5
รูปที่ 2.5 ตำแหน่งขาต่างๆ ของบอร์ด Node MCU V3 5
รูปที่ 2.6 จอ LCD 6
รูปที่ 2.7 Power supply 5 V 6
รูปที่ 3.1 ผังการดำเนินงานโครงงาน 7
รูปที่ 3.2 บล็อกไดอะแกรม พัดลมปรับความเร็วตามอุณหภูมิ 8
รูปที่ 3.4 แผนผังการทำงานของพัดลมปรับระดับความเร็วตามอุณหภูมิ 9
รูปที่ 3.5 writing diagram ของพัดลมปรับความเร็วตามอุณหภูมิ 10
รูปที่ 3.4 วางแผนออกแบบ 18
รูปที่ 3.5 การเขียนโปรแกรม 18
รูปที่ 3.6 ทดสอบวงจร 18
รูปที่ 3.7 นำแผงวงจรมาประกอบเขากับกลองใสอุปกรณ์ 19
รูปที่ 3.8 นำกลองอุปกรณ์ไปติดตั้งกับพัดลม 19
รูปที่ 3.9 ทดสอบการทำงานของพัดลมผ่าน Blynk 19

บทที่ 1
บทนำ

1.1 ที่มาและความสำคัญ

ในปัจจุบันสภาพอากาศของประเทศไทยมีอากาศร้อนจัดเนื่องจากเกิดภาวะโลกร้อน จึงทำให้คนส่วนใหญ่มีความจำเป็นต้องหาอุปกรณ์ในการคลายร้อน ซึ่งนอกเหนือจากเครื่องปรับอากาศทางคณะผู้จัดทำจึงได้คิดจะประดิษฐ์พัดลมปรับความเร็วตามอุณหภูมิด้วยระบบ ESP8266

ซึ่งคณะผู้จัดทำจะพูดถึงคือพัดลมที่ช่วยอำนวยความสะดวกในการใช้งานด้าน โดยปกติแล้วพัดลมที่มีการใช้งานอยู่กับในปัจจุบันจะเป็นแบบปกติและแม้ว่าการใช้พัดลมแทนเครื่องปรับอากาศจะช่วยลดการใช้พลังงานไฟฟ้าได้อย่างมาก แต่หากใช้อย่างไม่ประหยัดก็ยังถือว่าเป็นการสิ้นเปลืองพลังงานโดยเปล่าประโยชน์เช่นกัน โดยเฉพาะอย่างยิ่งการใช้พัดลมขณะนอนหลับพักผ่อนในเวลากลางคืน ซึ่งสภาพอากาศเปลี่ยนแปลงอยู่บ่อยๆและอุณหภูมิแต่ละองศาแตกต่างกันออกไปตามสภาพอากาศ ทำให้ร่างกายของเราปรับตัวตามสภาพอากาศไม่ทัน จึงทำให้ไม่สบายหรือเป็นหวัดอยู่บ่อยๆ ยิ่งสภาพอากาศตอนกลางคืนจะหนาวกว่ากลางวัน ซึ่งพัดลมที่เราใช้กันอยู่เวลาเรานอนหลับแล้วพัดลมยังทำงานอยู่ยิ่งดึกความเย็นยิ่งมากขึ้น แต่พัดลม ที่เราเปิดเอาไว้มันไม่สามารถเปลี่ยนระดับความเร็วลงตามอุณหภูมิของห้องนอนหรือสภาพอากาศได้นั่นเอง

1.2 วัตถุประสงค์

1.2.1 เพื่อศึกษาขั้นตอนวิธีการทำงานผ่านแอปพลิเคชั่น

1.2.2 เพื่อทดสอบประสิทธิภาพของพัดลมอัตโนมัติปรับระดับตามอุณหภูมิ

1.2.3 เพื่อนำความรู้ภาคทฤษฎีมาบูรณาการใช้ในการทำโครงงาน

1.3 ขอบเขตการทำโครงงาน

1.3.1 สามารถเชื่อมต่อกับสมาร์ทโฟน สั่งการผ่านสมาร์ทโฟนได้

1.3.2 สามารถดูสถานะอุณหภูมิความชื้นผ่านจอ LCD

1.3.3 อุณหภูมิต่ำกว่า 25°C พัดลมจะไม่ทำงานถ้าอุณหภูมิสูงกว่า 28°C จะทำงานที่เบอร์ 3

1.4 ประโยชน์ที่ได้รับจากโครงงาน

1.4.1 สร้างชุดควบคุมพัดลมปรับความเร็วตามระดับอุณหภูมิ

1.4.2 ได้ความรู้เกี่ยวกับการประดิษฐ์พัดลมอัตโนมัติปรับระดับตามอุณหภูมิ

1.4.3 เพิ่มความสะดวกสบายในการใช้งาน

บทที่ 2
ทฤษฎีที่เกี่ยวข้อง

การทำโครงงานนี้จะต้องศึกษาหลักการทำงานของอุปกรณ์อิเล็กทรอนิกส์ต่างๆ ให้เข้าใจอย่างละเอียดก่อนที่จะประกอบส่วนต่างเข้าด้วยกัน อุปกรณ์ที่สำคัญของโครงงานนี้คือ เซนเซอร์วัดอุณหภูมิ นอกจากนั้นจะต้องสามารถเขียนโปรแกรมวัดอุณหภูมิ ให้สามารถทำงานได้ตามวัตถุประสงค์ ต้องมีการวิเคราะห์และหาข้อมูลในส่วนๆนั้น เพื่อเป็นข้อมูลที่จะนำไปทำโครงงานและพัฒนาต่อ คณะผู้จัดทำได้ลำดับหัวข้อเพื่อศึกษาข้อมูลต่างๆ ดังนี้

2.1 เซ็นเซอร์วัดอุณหภูมิ และ ความชื้น DTH22

หลักการทำงาน

DHT22 เป็นโมดูลวัดอุณหภูมิและความชื้นที่ความละเอียดและช่วงการวัดที่สูงกว่า DHT11 ใช้ไฟได้ 3-5V สามารถวัดอุณหภูมิได้ตั้งแต่ -40 ถึง 80°C ที่ความแม่นยำ ±0.5°C และความชื้น 0-100% คลาดเคลื่อน 2-5% อัตราการอ่านค่าอุณหภูมิและความชื้นสูงสุด 0.5Hz

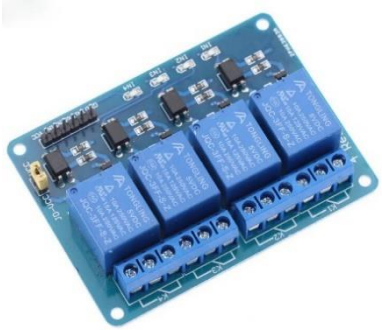
คุณสมบัติ

- ย่านวัดความชื่น 0 - 100% RH คลาดเคลื่อน+/- 5% RH
- ย่านวัดอุณหภูมิ -40 – 80% องศา ค่าความแม่นยำ +/- 5% องศาเซลเซียส
- มี 3 ขา
- ใช้ไฟเลี้ยง 3 - 5.5 VDC
- อ่านค่าสัญญาณทุก 1 วินาที

รูปที่ 2.1 DTH22

ที่มา : https://www.arduino4.com

2.2 รีเลยโมดูล (Relay Module)



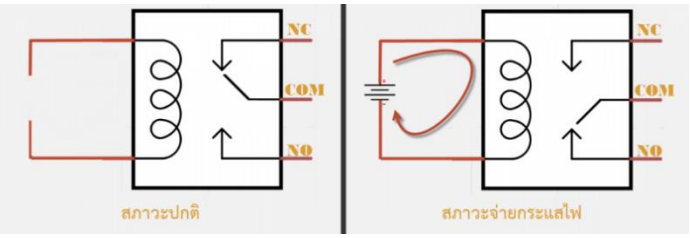
รูปที่ 2.2 รีเลย์บอร์ด (Relay Board)

ที่มา : https://www.ai-corporation.net

โมดูลรีเลย์ 4ช่อง 5V (4 Channel Relay Module) เป็นโมดูลที่ใช้ควบคุมการทำงานของโหลดทางไฟฟ้าได้ทั้งแรงดันไฟฟ้ากระแสตรง (DC) และไฟฟ้ากระแสสลับ (AC) ซึ่งโหลดสูงสุด คือ AC 250V/10A และ DC 30V/10A โดยใช้สัญญาณในการควบคุมการทำงานด้วยสัญญาณลอจิก TTL ทำงานด้วยสัญญาณแบบ Active Low,กระแสขับรีเลย์ (Drive Current) 15-20mA., มีการออกแบบให้เป็นIsolate ด้วย Optocoupler, มี LED แสดงสถานะ Relayสามารถนำไปประยุกต์ใช้งานPLC Control,บ้านอัจฉะริยะใช้ในโรงงานอุตสาหกรรมหรืองานอื่นๆ ขึ้นอยู่กับการเขียนโปรแกรมและการต่อใช้งานภายนอก สามารถเชื่อมต่อใช้งานกับบอร์ด Rasberry Pi, Arduino, NodeMCU ESP8266, NodeMCU ESP8266 ฯลฯ เป็นต้น

คุณสมบัติ

- ไฟเลี้ยงโมดูลรีเลย์ VCC = 5VDC
- ควบคุมโหลดได้ทั้งแรงดันไฟฟ้า AC ได้สูงสุด 250VAC 10A หรือ แรงดันไฟฟ้า DC ได้สูงสุด 30VDC 10A (Maximum Load)
- ระดับสัญญาณอินพุตควบคุมแบบ TTL ทำงานด้วยสัญญาณแบบ Active Low
- กระแสขับรีเลย์ (Drive Current) 15-20mA
- มีการออกแบบให้เป็น Isolate ด้วย Optocoupler
- มี LED แสดงสถานะ Relay
- โมดูลขนาด 5.3cm.(กว้าง) x 7.0cm.(ยาว) x 1.7cm.(สูง)



รูปที่ 2.3 แสดงสภาวะการทำงานของรีเลย์

ที่มา : https://www.ai-corporation.net

จุดต่อใช้งานมาตรฐาน

จุดต่อ NC (normal closed) หมายความว่าปิดปกติหรือหากยังไม่จ่ายไฟให้ขดลวดเหนี่ยวนำหน้าสัมผัสจะต่อถึงกันโดยทั่วไปเรามาต่อจุดนี้เข้ากับอุปกรณ์หรือเครื่องใช้ไฟฟ้าที่ต้องการให้ทำงานตลอดเวลา

จุดต่อ NO (normal opened) หมายความว่าปกติเปิดหรือหากยังไม่จ่ายไฟฟ้าขดลวดเหนี่ยวนำหน้าสัมผัสจะไม่ต่อถึงกันกันทั่วไป เรามักต่อจุดนี้เข้ากับอุปกรณ์หรือเครื่องใช้ไฟฟ้าที่ต้องการควบคุมการเปิดปิด

จุดต่อ COM ย่อมาจาก common คือจุดร่วมที่ต่อมาจากแหล่งจ่ายไฟ

2.3 บอร์ดไมโครคอนโทรลเลอร์ Node MCU ESP8266

Node MCN เป็นบอร์ดที่ใช้ ESP8266 เป็น CPU สำหรับประมวลผลโปรแกรมต่างๆมีข้อดีกว่า Arduino ตรงที่ตัวมันมีขนาดเล็กกว่ามีพื้นที่เขียนโปรแกรมลงไปมากกว่าและสามารถเชื่อมตัวกับ Wi-Fi ได้บนบอร์ดรุ่นนี้ใช้ ESP8266 12E มีพื้นที่หน่วยความจำรอมสูงถึง 4MB เพียงพอสำหรับการเขียนโปรแกรมขนาดใหญ่ อีกทั้งภายในยังเป็น ARM ขนาดย่อมๆ ใช้ความถี่สูงถึง 40MHz ทำให้สามารถประมวลผลโค้ดโปรแกรมได้อย่างรวดเร็วเหมาะมากสำหรับงาน Smart Home และ IOT

ESP8266 เป็นชื่อเรียกของชิฟและของโมดูล ESP8266 สำหรับติดต่อสื่อสารบนมาตรฐาน Wi-Fi ทำงานที่แรงดันไฟฟ้า 3.0 - 3.6 V ทำงานใช้กระแสโดยเฉลี่ย 80mA รองรับคำสั่ง deep sleep ในการประหยัดพลังงานโดยใช้กระแสน้อยกว่า 10 ไมโครแอมป์ สามารถ wake up กลับมาส่งข้อมูลใช้เวลาน้อยกว่า 2 มิลลิวินาที ภายในมี Low power MCU 32bit ทำให้เราเขียนโปรแกรมสั่งงานได้ มีวงจร analog to digital converter ทำให้สามารถอ่านค่าจาก analog ได้ความละเอียด 10-bit ทำงานได้ที่อุณหภูมิ -40 ถึง 125 องศาเซลเซียส ESP8266 และสามารถติดต่ออุปกรณ์อื่นๆหรือเซ็นเซอร์ต่างๆในแบบของ Arduino ถ้ามีพื้นฐาน Arduino อยู่แล้วก็จะเข้าใจและใช้งานได้รวดเร็ว

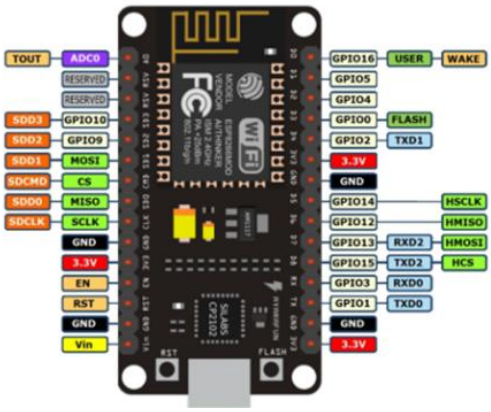
2.3 การเขียนโปรแกรม ESP8266 Arduino IDE

สำหรับการต่อวงจรของ ESP8266เพื่อเขียนโปรแกรมด้วย Arduino เขียนโค้ดเหมือนกันทุกรุ่นโดยแต่ละรุ่นจะมีขาไม่เท่ากัน ดังนั้นเราจึงต้องเปรียบเทียบขา GPIO ให้ถูกต้องในการสั่งงานก็ใช้ได้แล้วการเขียนโปรแกรมอัปโหลดโค้ดลงบอรด ESP8266 เกือบทุกรุ่น จะผานทาง Serial ที่ขา rx,tx โดยใช้โมดูลUSB,TTL ซึ่งต้องเสียเวลาในการต่อวงจรเพื่ออัปโหลดโค้ดอีกทั้งโมดูล ESP8266 หลายๆรุ่นมีการต่อขาที่เป็นแบบเซอร์เฟสเมาส์ทำให้ไม่สะดวกกับการต่อทดลองบนบอร์ดทดลองดังนั้นจึงมีการรวมโมดูลUSB,TTLและต่อวงจรขยายขา ESP8266 ให้เป็นขาระยะห่างขนาด 2.54 mm ซึ่งสามารถเสียบลงบอร์ดทดลอง ได้พอดีกลายเป็นบอร์ด ESP8266 โดยหนึ่งในบอร์ดที่นิยมใช้งานคือ Node MCU ซึ่งใชโมดูล ESP8266 ESP-12E

บอร์ด ESP8266 Node MCU V3

รูปที่ 2.4 Node MCU Version3

ที่มา : https://images.app.goo.gl/r346chsGuRhYBR9w9



รูปที่ 2.5 ตำแหน่งขาต่างๆ ของบอ์ด Node MCU V3

ที่มา : https://images.app.goo.gl/EypKKWCN9uaJ8CZu9

2.4 จอ LCD

รูปที่ 2.6 จอ LCD

ที่มา : https://static.cytron.io/image/cache/catalog/products/DS-162A-I2C-G/DS-162A-I2C-G%20%20(1)-800x800.jpg

รายละเอียด

- I 2 C ที่อยู่: 0x27 (ค่าเริ่มต้น)
- ไฟพื้นหลัง (ตัวอักษรสีดำบนพื้นหลังสีเหลือง/เขียว) สามารถควบคุมได้ผ่านมินิจั๊มเปอร์
- แรงดันไฟ: 5V
- ขนาด : 82x35x18 มม.
- มาพร้อมกับอินเตอร์เฟส I 2 C

2.5 Power supply 5 V

แหล่งจ่ายไฟ 5 V ให้อุปกรณ์อิเล็กทรอนิกส์ เสียบเขาบอรด Arduino ไดโดยตรง

รูปที่ 2.7 Power supply 5 V

ที่มา: https://th.rs-online.com/web/p/switching-power-supplies/6446957

รายละเอียด

-Input Voltage: 110/220V AC

-Output Current: 5A

-Frequency: 47 ~ 63Hz

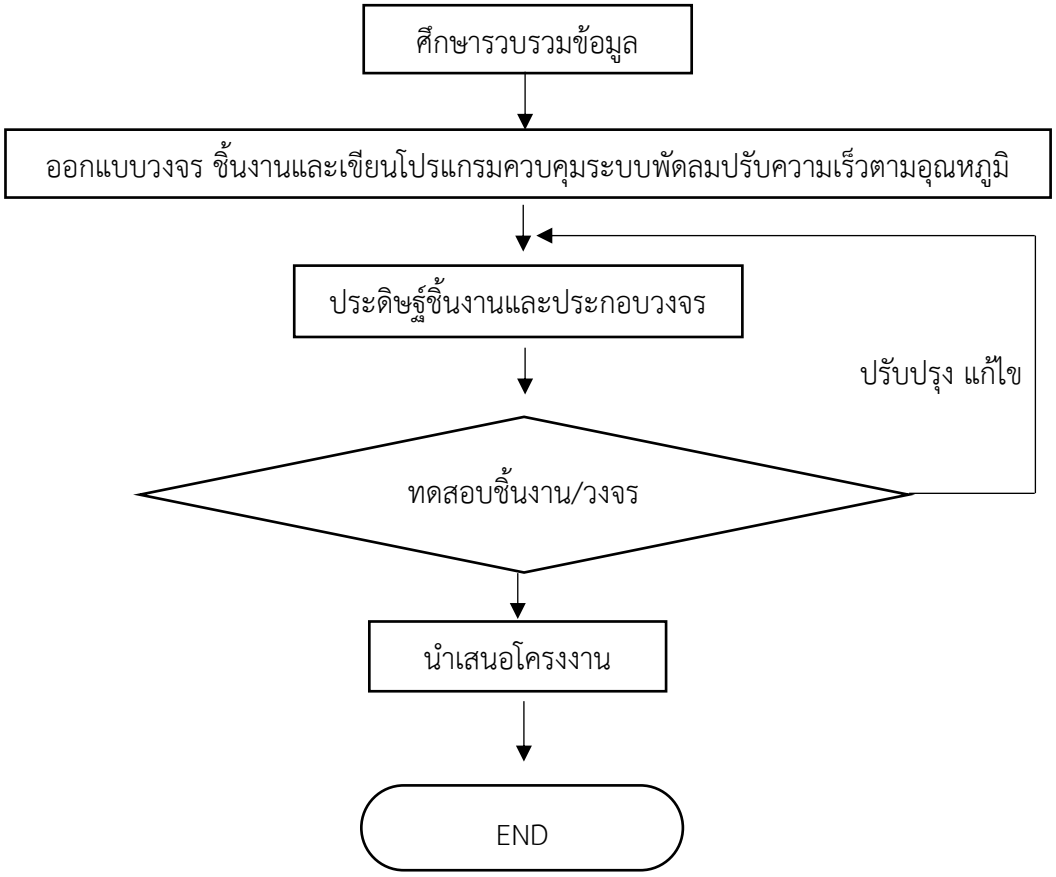
- Output Voltage:DC 5V

- Power: 25W

- Size: 85 * 58 * 35MM

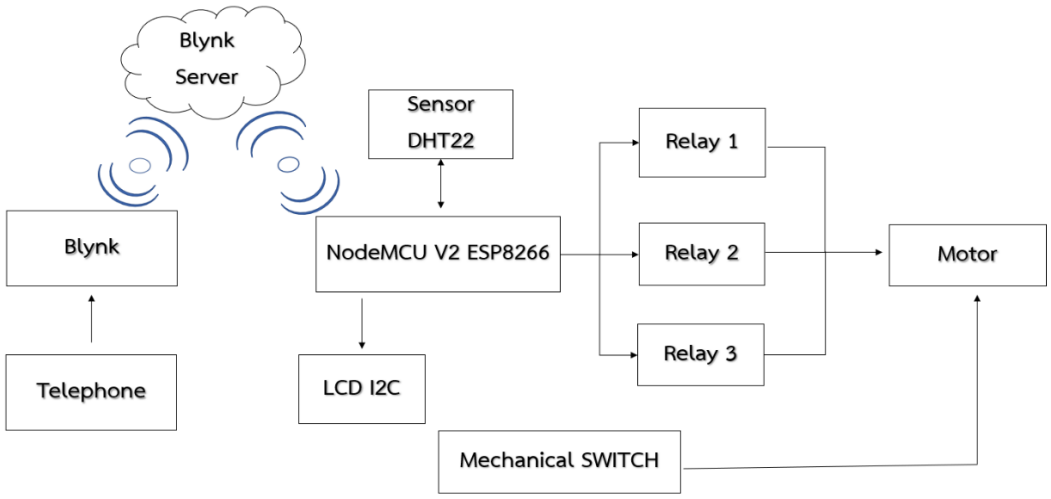
บทที่ 3
การดำเนินงาน

3.1 วิธีการดำเนินงาน



รูปที่ 3.1 ผังการดำเนินงานโครงงาน

3.2 บล็อกไดอะแกรม



รูปที่ 3.2 บล็อกไดอะแกรม พัดลมปรับความเร็วตามอุณหภูมิ

ส่วนประกอบของบล็อกไดอะแกรม

Blynk => แอปที่ใช้สั่งการพัดลมทำงานระยะไกล

Blynk sever => ตัวกลางในการสื่อสาร

Telephone => จะทำหน้าที่คล้ายกับรีโมตควบคุมการทำงานผ่าน wireless โดยใช้ Blynk App เป็นตัวควบคุม

NodeMCU V2 ESP8266 => หัวใจสำคัญในการควบคุมการทำงานของวงจร

LCD I2C => แสดงผลของอุณหภูมิ ณ ช่วงนั้น เป็นหน่วยองศาเซลเซียส

Sensor DHT22 => วัดอุณหภูมิทุกๆ1วินาที

Relay1 , Relay2 และ Relay3 => เปรียบเสมือนสวิตช์พัดลมโดยควบคุมด้วยบอร์ด ESP8266

Switch => สวิตช์พัดลมแบบปกติ

Motor => เป็นมอเตอร์ของพัดลมที่ทำงานตามขดลวด 3 ขด

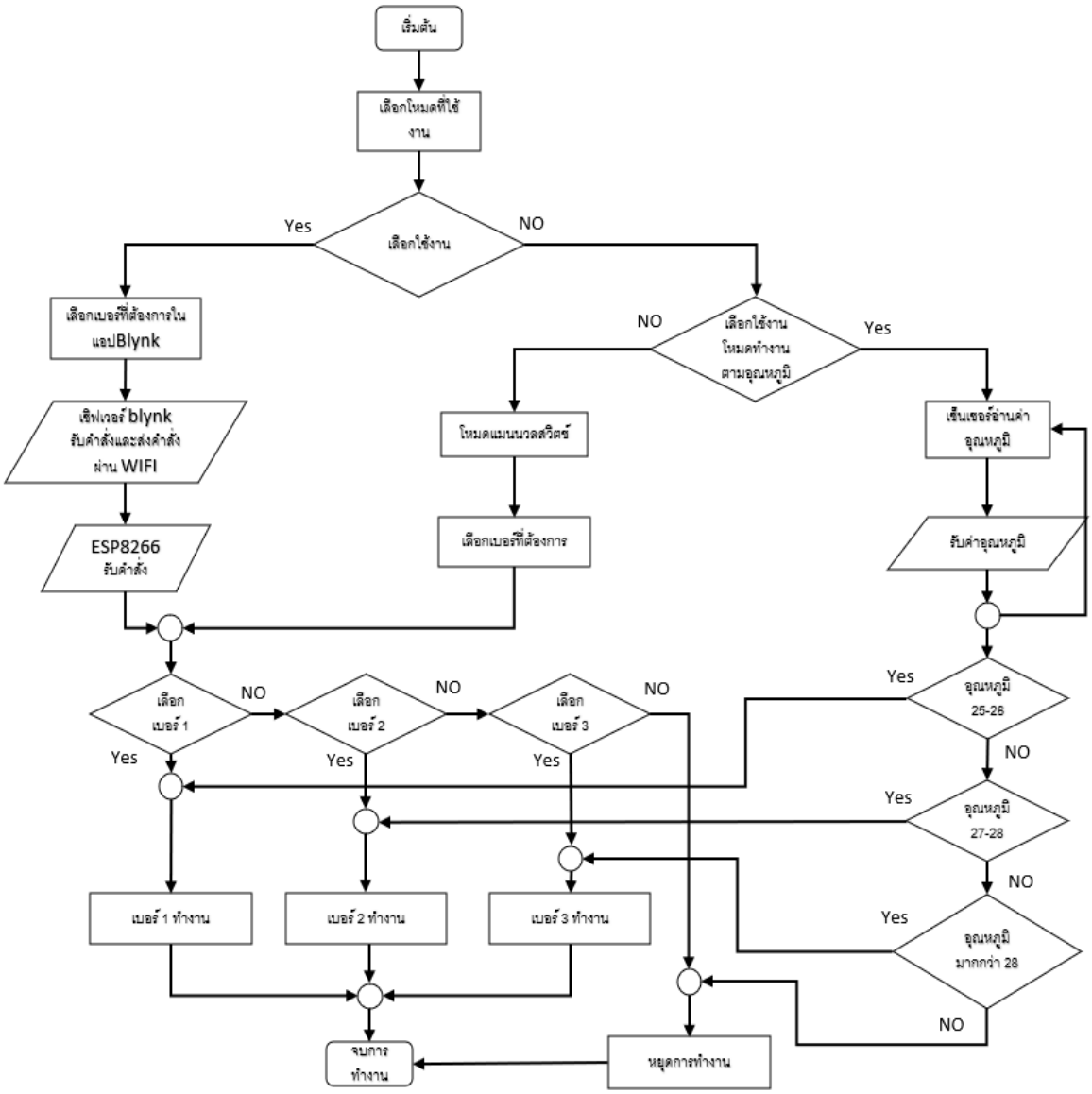
พัดลมมีการทำงานทั้งหมด 3 โหมด ได้แก่โหมดออโต้ ไร้สายและแบบปกติ

หลักการทำงานของพัดลม

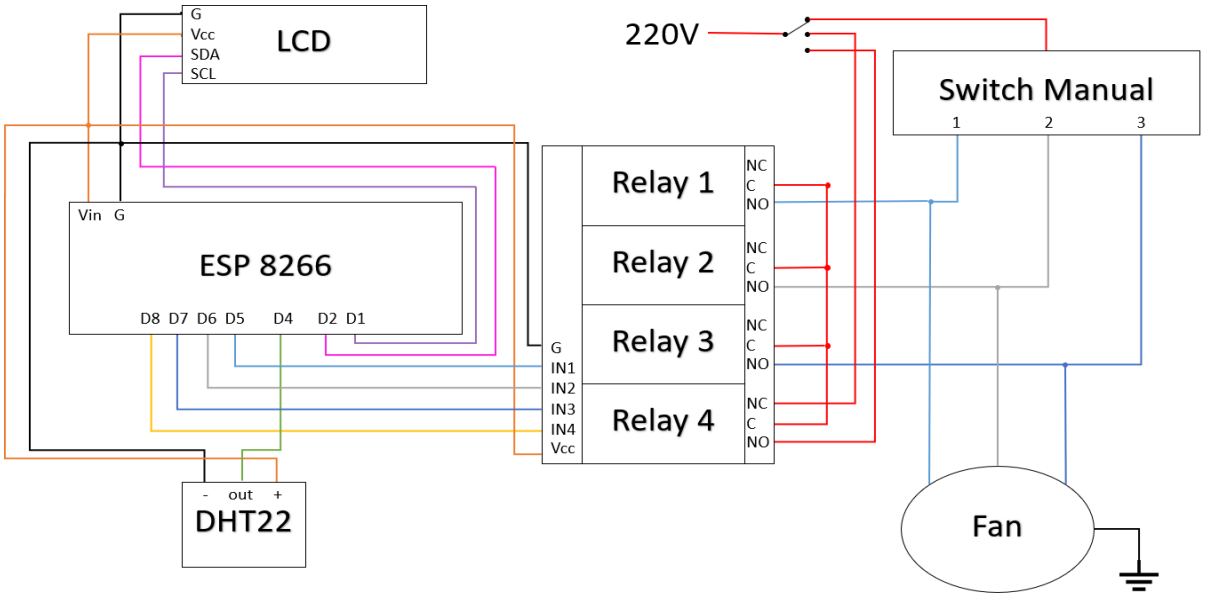
1. โหมดออโต้ เซ็นเซอร์จะคอยวัดอุณหภูมิเพื่อส่งข้อมูลมาที่บอร์ดเพื่อไปควบคุมการทำงานของพัดลมและไปแสดงค่าอุณหภูมิ ณ ขณะนั้นที่จอ LCD

2.โหมดไร้สาย โทรศัพท์จะทำหน้าที่คล้ายกันกับรีโมตควบคุมการทำงานของพัดลม โดยผ่านทาง Blynk app จะส่งข้อมูลให้ที่บอร์ดเพื่อควบคุมการทำงานของพัดลมผ่านทาง wireless

3.โหมดปกติ สามารถมากดที่สวิตช์พัดลมได้ตามปกติ



รูปที่ 3.4 แผนผังการทำงานของพัดลมปรับระดับความเร็วตามอุณหภูมิ



รูปที่ 3.5 writing diagram ของพัดลมปรับความเร็วตามอุณหภูมิ

ขา Vin ของบอร์ดESP8266 ต่อกับขา Vcc ของ LCD, Relay และ ขา + ของ Sensor DHT22

ขา G ของบอร์ดESP8266 ต่อกับขา G ของ LCD, Relay และ ขา - ของ Sensor DHT22

ขา D1 ของบอร์ดESP8266 ต่อกับขา SCL ของ LCD

ขา D2 ของบอร์ดESP8266 ต่อกับขา SDA ของ LCD

ขา D4 ของบอร์ดESP8266 ต่อกับขา Out ของ Sensor DHT22

ขา D5 ของบอร์ดESP8266 ต่อกับขา IN1 ของ Relay

ขา D6 ของบอร์ดESP8266 ต่อกับขา IN2 ของ Relay

ขา D7 ของบอร์ดESP8266 ต่อกับขา IN3 ของ Relay

ขา D8 ของบอร์ดESP8266 ต่อกับขา IN4 ของ Relay

ไฟ 220V ต่อเข้าขา C ของสวิตช์ 4 ขา ขาแรกต่อไปยัง สวิตช์แมนนวล ขาที่สองต่อไปยังขา NC ของ Relay4 ขาที่สามต่อไปยังขา NO ของ Relay4

ขา NO ของ Relay1 และ สวิตช์เบอร์1 ต่อเข้าหากันแล้วต่อไปหาพัดลมเบอร์1

ขา NO ของ Relay2 และ สวิตช์เบอร์2 ต่อเข้าหากันแล้วต่อไปหาพัดลมเบอร์2

ขา NO ของ Relay3 และ สวิตช์เบอร์3 ต่อเข้าหากันแล้วต่อไปหาพัดลมเบอร์3

3.2 แผนงานในการดำเนินโครงการ

ลำดับ	ขั้นตอนการดำเนินงาน	ระยะเวลาการดำเนินงาน												หมายเหตุ
		30 ม.ค. - 24 มี.ค. 2566												
		ม.ค.				ก.พ.				มี.ค.				
		1	2	3	4	1	2	3	4	1	2	3	4	
1	เสนอโครงงาน													
2	ศึกษาข้อมูลโครงงาน													
	2.1 แบ่งหน้าที่ของแต่ละบุคคล													
	2.2 จัดหาอุปกรณ์													
3	จัดเตรียมอุปกรณ์ชิ้นงาน													
	3.1 ออกแบบ Block Diagram													
	3.2 ออกแบบ Writing Diagram													
	3.3 ประดิษฐ์ชิ้นงานที่ออกแบบไว้													
	3.4 ทดสอบชิ้นงาน													
4	จัดทำรูปเล่ม													
	4.1 บทที่1-3													
	4.2 บทที่4-5													
5	ติดตามความก้าวหน้าโครงงาน													
6	ฝึกเสนอโครงงาน													
7	ส่งชิ้นงาน และเอกสารโครงงาน													
8	สอบโครงงาน													
9	ถ่ายวีดีโอนำเสนอโครงงาน													
10	จัดทำบอร์ดนิทรรศการโครงงาน													
11	จัดนิรรศการโครงงาน													

3.3 รายการวัสดุอุปกรณ์ที่ใช้ในการทำโครงงานสิ่งประดิษฐ์

รายการอุปกรณ์ในการทำโครงงานสิ่งประดิษฐ์				
ลำดับ	**รายการ**	**จำนวน**	**หน่วย**	**ราคา(บาท)**
1	เซ็นเซอร์วัดอุณหภูมิและความชื่น DHT22	1	บอร์ด	60
2	รีเลย์ 4 ช่อง	1	ตัว	140
3	บอร์ด ESP8266	1	ตัว	270
4	สายจั๊มป์เปอร์	1	ชุด	100
5	สายไฟ	1	เมตร	50
6	พัดลมขนาด 16 นิ้ว	1	ตัว	-
7	MCU Base	1	ตัว	100
8	จอLED ขนาด 16*2	1	ชิ้น	50
9	พาเวอร์ซัพพลาย 5 V	1	ตัว	145
10	กล่องสำหรับใส่วงจร	1	กล่อง	105
11	กระป๋องสีสำหรับรองพื้นและพ่นสี	2	กระป๋อง	-
	รวม			1,020

บทที่ 4

ผลการทดลอง

จากการทำโครงงานสิ่งประดิษฐ์นี้สมาชิกกลุ่มได้ทำการสำรวจความต้องการของนักเรียนจ่าโรงเรียนอิเล็กทรอนิกส์ ในความต้องการของระดับพัดลมว่า แต่ละสภาพอากาศแต่ละอุณหภูมินักเรียนจ่าโรงเรียนอิเล็กทรอนิกส์ มีความต้องการพัดลมอยู่ที่เบอร์อะไร เช่น อุณหภูมิต่ำต้องการพัดลมอยู่ที่เบอร์1 อุณหภูมิสูงต้องการอยู่ที่เบอร์3 ซึ่งความต้องการของนักเรียนไม่เหมือนกันจึงสรุปมาดังตาราง และทำการทดลองของพัดลมตามอุณหภูมิว่าสามารถทำงานได้ตามที่เราต้องการหรือไม่

4.1 ตารางแบบสอบถามความต้องการร

ตารางแบบสอบความต้องการของระดับลม กับนักเรียนจ่าโรงเรียนอิเล็กทรอนิกส์		
ลำดับ	**อุณหภูมิ (องศา)**	**ความต้องการของระดับพัดลม(เบอร์)**
1	30	เบอร์3
2	24	ไม่ต้องการ
3	26	เบอร์1
4	25	เบอร์1
5	28	เบอร์3
6	24	ไม่ต้องการ
7	27	เบอร์2
8	27	เบอร์2
9	23	ไม่ต้องการ
10	32	เบอร์3

ตารางที่ 4.1 แบบสอบถามความต้องการ

จากการตรวจสอบความต้องการของนักเรียนจ่าโรงเรียนอิเล็กทรอนิกส์ จำนวน 10 นาย ความต้องการความแรงของระดับพัดลมเบอร์ 3 มีจำนวน 3 นาย ความต้องการเบอร์ 2 มีจำนวน 2 นาย ความต้องการเบอร์ 1 มีจำนวน 2 นาย ไม่ต้องการให้พัดลมทำงาน มี 3 นาย

อุณหภูมิ (องศา)	ระดับพัดลม (เบอร์)	จำนวน(นาย)
ต่ำกว่า 25	0	3
25-26	1	2
26-27	2	2
มากกว่า 28	3	3

ตารางที่ 4.2 สรุปแบบสอบถาม

4.2 การทดลองพัดลมปรับความเร็วตามอุณหภูมิและผ่าน Blynk

การออกแบบและจัดทำพัดลมควบคุมอุณหภูมิคณะผู้จัดทำได้ทำการทดสอบเพื่อหาสมรรถณะของพัดลมควบคุมและอุณหภูมิโดยการทดสอบหาประสิทธิภาพการทำงานโดยมีการออกแบบและสร้างขึ้นได้ตามวัตถุประสงค์และขีดความสามารถที่ตั้งไว้และผลการทดสอบที่ได้จากการออกแบบและสร้างขึ้นคือส่วนประกอบของพัดลมควบคุมด้วยอุณหภูมิโดยมีผลการทดสอบในการทำงานของพัดลมควบคุมด้วยอุณหภูมิดังนี้

ระดับความเร็ว (เบอร์)	อุณหภูมิ	ผล
0	ต่ำกว่า 25 องศา	พัดลมหยุดทำงาน
1	25-26 องศา	พัดลมทำงาน
2	27-28 องศา	พัดลมทำงาน
3	มากกว่า 28 องศา	พัดลมทำงาน

ตารางที่ 4.3. สรุปผลการทดลองผ่านอุณหภูมิ

สรุปผลการทดลอง

เมื่อเปิดพัดลมปรับความตามระดับอุณหภูมิ พัดลมจะปรับตามอุณหภูมิในห้อง จอLCD จะแสดงอุณหภูมิตามที่เซ็นเซอร์วัดได้ ถ้าอุณหภูมิในห้องต่ำกว่า 25 องศา จะหยุดทำงานอัตโนมัติ เมื่ออุณหภูมิเพิ่มขึ้นเป็น 25-26 องศาพัดลมจะทำงานที่เบอร์ 1 แต่ถ้าอุณหภูมิเพิ่มขึ้นเป็น 27-28 องศาพัดลมจะทำงานที่เบอร์ 2 และตั้งแต่ 28 องศา ขึ้นไปพัดลมจะทำงานที่เบอร์ 3

ผลการทดลองผ่าน แอพ BLYNK

สวิตช์(เบอร์)	พัดลม	
	ทำงาน	ไม่ทำงาน
0		
1		
2		
3		

ตารางที่ 4.4 สรุปผลกาทดลองด้วยสวิตช์

สรุปผลการทดลองผ่าน แอพ BLYNK

สวิตช์ที่ 1 ทำงาน

สวิตช์ที่ 2 ทำงาน

สวิตช์ที่ 3 ทำงาน

บทที่ 5
สรุปปัญหาและข้อเสนอแนะ

5.1 สรุปโครงงาน

จากผลการดำเนินโครงงานสิ่งประดิษฐ์เรื่องพัดลมปรับความเร็วตามระดับอุณหภูมิและสั่งการผ่านแอป Blynk ถูกใช้เพื่อเพิ่มความสะดวกสบายในการใช้งานภายในบ้าน โรงเรียน โรงพยาบาล ดังนั้นในการจัดทำโครงงานนี้ ใช้เป็นแนวทางการศึกษา ทำให้ได้เรียนรู้ถึงวิธีการทำงานของพัดลมตามอุณหภูมิที่เราเขียนไว้ในโปรแกรม ทั้งนี้โครงงานสำฤทธิ์ผลได้จากการทำงานเป็นหมู่คณะ ตลอดจนการร่วมกันคิดแก้ไขปัญหาข้อบกพร่องและสามารถนำไปต่อยอดได้ในอนาคต

5.2 ปัญหา

1. ในกรณีที่บอร์ด ESP8266 ไม่มีไฟเลี้ยงจะทำให้ไม่สามารถสั่งการทำงานของระบบได้
2. บอร์ด ESP8266 จำเป็นที่จะเชื่อมต่ออินเทอร์เน็ตตลอดเวลาเพื่อที่จะสั่งการปรับความเร็วใบพัด
3. ปัญหาการเลือกอุปกรณ์การขาดความเข้าใจในอุปกรณ์ที่จะนำมาใช้
4. ปัญหาการตรวจสอบอุปกรณ์ว่าทำงานได้ไม่เต็มประสิทธิภาพจนไปถึงไม่สามารถทำงานได้ซึ่งทำให้เป็นอุปสรรคในการหาอุปกรณ์มาทดแทนอุปกรณ์ที่ไม่สามารถใช้งานได้หรือได้ไม่ตรงกับที่ต้องการ
5. บอร์ด ESP 8266 ไม่ทำงานเนื่องจากลงโค้ดไม่ตรงตามที่ต้องการ
6. ในระหว่างที่ใช้โหมดออโต้แล้วเปลี่ยนไปใช้โหมดอื่นนั้น หลังจากที่เราจะกลับมาใช้โหมดออโต้อีกครั้งพบว่าเซ็นเซอร์จะหยุดการทำงาน

5.3 ข้อเสนอแนะ

ปัญหาที่เกิดขึ้นในการจัดทำโครงงานชิ้นนี้ พบในสวนของเครื่องตนแบบ ซึ่งทำให้เกิดข้อผิดพลาดในการทำงานของระบบการทำงาน ผู้จัดทำจึงขอเสนอแนวทางในการแก้ปัญหา ดังตอไปนี้

1. การเลือกวัสดุอุปกรณ์ทาง รร.อล.กวก.อล.ทร. ควรจะเปิดโอกาสให้นักเรียนได้ศึกษาทดสอบอุปกรณ์ที่ต้องใช้ในการจัดทำโครงงาน รวมถึงทฤษฎีการทำงาน วิธีการติดตั้ง รวมทั้งขนาดที่เหมาะสมเพื่อที่จะเลือกใช้อุปกรณ์ได้อย่างถูกต้อง
2. โปรแกรม Arduino นั้นมีความซับซ้อน ผู้ศึกษาต้องใช้ความมุ่งมั่น และความพยายาม เพื่อทำความเข้าใจอย่างลึกซึ้งในการศึกษาโปรแกรม รวมไปถึงต้องทุ่มเทให้เวลาในการศึกษาโปรแกรม

บรรณานุกรม

โครงงานสิ่งประดิษฐ์นักเรียนจ่ารุ่น 64 พัดลมปรับระดับความเร็วตามอุณหภูมิและผ่าน Blynk

1.สอนใช้งาน ESP8266 เริ่มต้นใช้งาน ติดตั้งโปรแกรมเบื้องต้น ESP8266 สืบค้นเมื่อวันที่ 10 มีนาคม 2566 จากเว็บไซต์ : https://www.cybertice.com/article/79/สอนใช้งาน-nodemcu-esp8266--nodemcu-esp8266-เริ่มติดตั้ง-nodemcu-esp8266-ลงบน-arduino-ide

2.สอนใช้งาน Arduino ESP8266 ด้วย Wifi สืบค้นเมื่อวันที่ 10 มีนาคม 2566 จากเว็บไซต์ : https://www.cybertice.com

3.สอนใช้งาน DHT22 Module โมดูลวัดอุณหภูมิและความชื้น กับ Arduino สืบค้นเมื่อวันที่ 10 มีนาคม 2566 จากเว็บไซต์ : https://www.cybertice.com/article/64/สอนใช้งาน-dht22-module-โมดูลวัดอุณหภูมิและความชื้น-กับ-arduino

4.รีเลย์ คืออะไร มีหลักการทำงานอย่างไรบ้าง สืบค้นเมื่อวันที่ 10 มีนาคม 2566 จากเว็บไซต์ : https://misumitechnical.com/technical/electrical/relay-working-principles/

5.สอนใช้งานจอ LCD เข้ากับบอร์ด arduino สืบค้นเมื่อวันที่ 10 มีนาคม2566 จากเว็บไซต์ : : https://www.cybertice.com/article/47/การใช้งานจอ-character-lcd-กับ-arduino

ภาคผนวก

ขั้นตอนการดำเนินงาน

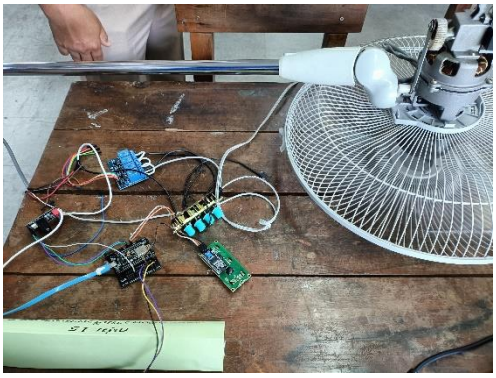
1. วางแผนออกแบบพัดลมปรับความเร็วตามอุณหภูมิและผ่าน Blynk

รูปที่ 3.4 วางแผนออกแบบ

2. เริ่มการเขียนโปรแกรมพัดลมปรับความเร็วตามอุณหภูมิและผ่าน Blynk

รูปที่ 3.5 การเขียนโปรแกรม

3. ทำการทดสอบวงจรพัดลมปรับความเร็วตามอุณหภูมิและผ่าน Blynk



รูปที่ 3.6 ทดสอบวงจร

4. นำแผงวงจรมาประกอบเข้ากับกลองใสอุปกรณ์

รูปที่ 3.7 นำแผงวงจรมาประกอบเขากับกลองใสอุปกรณ์

5. ทดลองนำกลองอุปกรณไปติดตั้งกับพัดลม

รูปที่ 3.8 นำกลองอุปกรณไปติดตั้งกับพัดลม

3.6.6 ทำการทดสอบการทำงานของพัดลมผ่าน Blynk

รูปที่ 3.9 ทดสอบการทำงานของพัดลมผ่าน Blynk

โปรแกรมพัดลมปรับระดับความเร็วตามอุณหภูมิและผ่าน Blynk

โปรแกรมการทำงาน

```
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
#include <Adafruit_Sensor.h>
#include <LiquidCrystal_I2C.h>
#include "DHT.h"

#define BLYNK_PRINT Serial
#define APP_DEBUG
#define DHTPIN 2
#define DHTTYPE DHT22
#define relay1 D5
#define relay2 D6
#define relay3 D7
#define relay4 D8

char auth[] = "B6QMcOhJBYxeOiQ_uGyMS99OU-GrelqO";
char ssid[] = "DADDY";
char pass[] = "19191199";
unsigned long period = 500;
unsigned long last_time = 0;

LiquidCrystal_I2C lcd(0x27, 16, 2);
DHT dht(DHTPIN, DHTTYPE);
BlynkTimer timer;

BLYNK_WRITE(V0) {
  Serial.println(param.asInt());
  if (param.asInt()) {
    digitalWrite(relay1, LOW);
    digitalWrite(relay2, HIGH);
    digitalWrite(relay3, HIGH);
```

```
    digitalWrite(relay4, HIGH);
    Blynk.virtualWrite(V1, 0);
    Blynk.virtualWrite(V2, 0);
  } else {
    digitalWrite(relay1, HIGH);
    digitalWrite(relay4, LOW);
  }
}
BLYNK_WRITE(V1) {
  Serial.println(param.asInt());
  if (param.asInt()) {
    digitalWrite(relay2, LOW);
    digitalWrite(relay1, HIGH);
    digitalWrite(relay3, HIGH);
    digitalWrite(relay4, HIGH);
    Blynk.virtualWrite(V0, 0);
    Blynk.virtualWrite(V2, 0);
  } else {
    digitalWrite(relay2, HIGH);
    digitalWrite(relay4, LOW);
  }
}
BLYNK_WRITE(V2) {
  Serial.println(param.asInt());
  if (param.asInt()) {
    digitalWrite(relay3, LOW);
    digitalWrite(relay1, HIGH);
    digitalWrite(relay2, HIGH);
    digitalWrite(relay4, HIGH);
    Blynk.virtualWrite(V0, 0);
    Blynk.virtualWrite(V1, 0);
  } else {
    digitalWrite(relay3, HIGH);
```

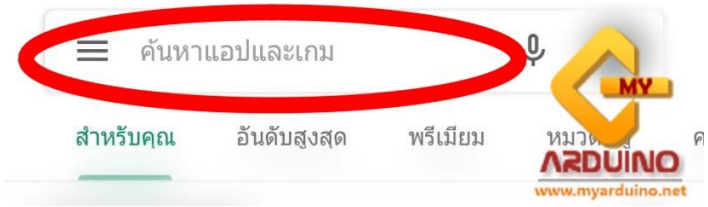
```
    digitalWrite(relay4, LOW);
  }
}

void setup() {
  Serial.begin(115200);
  Blynk.begin(auth, ssid, pass);
  pinMode(relay1, OUTPUT);
  pinMode(relay2, OUTPUT);
  pinMode(relay3, OUTPUT);
  pinMode(relay4, OUTPUT);
  digitalWrite(relay1, HIGH);
  digitalWrite(relay2, HIGH);
  digitalWrite(relay3, HIGH);
  digitalWrite(relay4, LOW);
  lcd.begin();
  dht.begin();
}
void loop() {
  if (millis() - last_time > period) {
    last_time = millis();
    Blynk.run();
    timer.run();
    float t = dht.readTemperature();
    float h = dht.readHumidity();
    Serial.print("Temp: ");
    Serial.print(t);
    Serial.println(" C");
    Serial.print("Humidity: ");
    Serial.print(h);
    Serial.println(" %");
    lcd.setCursor(0, 0);
    lcd.print("Hum:      ");
```

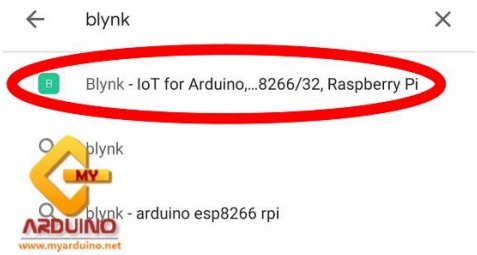
```
    lcd.setCursor(4, 0);
    lcd.print(h);
    lcd.setCursor(9, 0);
    lcd.print("%");
    lcd.setCursor(0, 1);
    lcd.print("Tem:      ");
    lcd.setCursor(4, 1);
    lcd.print(t);
    lcd.setCursor(9, 1);
    lcd.print("C");
    if (t < 25) {
      digitalWrite(relay2, HIGH);
      digitalWrite(relay1, HIGH);
      digitalWrite(relay3, HIGH);
    } else if (t >= 25 && t < 27) {
      digitalWrite(relay1, LOW);
      digitalWrite(relay2, HIGH);
      digitalWrite(relay3, HIGH);
    } else if (t >= 27 && t < 29) {
      digitalWrite(relay2, LOW);
      digitalWrite(relay1, HIGH);
      digitalWrite(relay3, HIGH);
    } else if (t >= 29) {
      digitalWrite(relay3, LOW);
      digitalWrite(relay1, HIGH);
      digitalWrite(relay2, HIGH);
    }
  }
}
```

เริ่มต้นใช้งาน Blynk

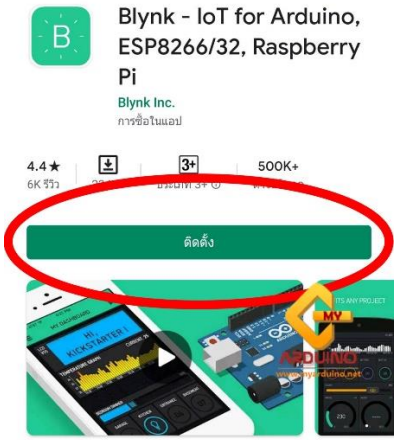
1.เปิดแอพฯ Play Store หรือ App Store ขึ้นมา



2.ไปที่ช่องค้นหาแล้วพิมพ์คำว่า blynk ลงไป แล้วเลือกการค้นหาแรก

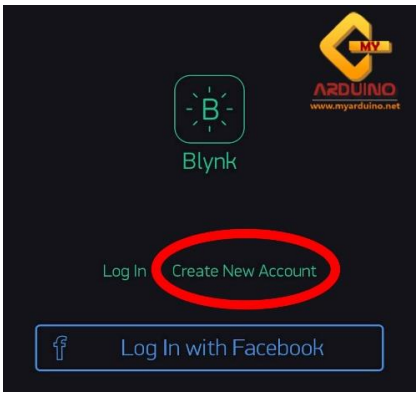


3.ทำการติดตั้งแอพฯ

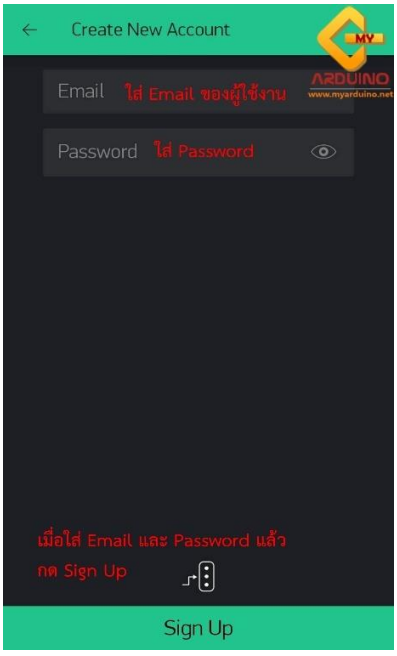


4.เปิดแอพฯขึ้นมา

5.เมื่อเปิดแอพฯขึ้นมาแล้วจะปรากฏหน้าจอตามรูปข้างล่าง สำหรับผู้ใช้งานครั้งแรกให้เลือก “Create New Account”

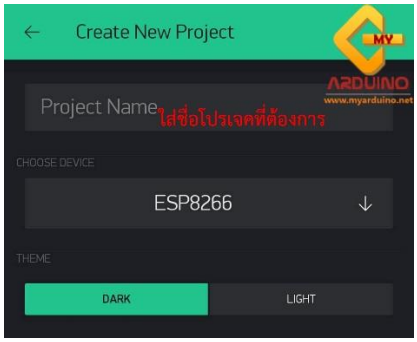


6.ให้ผู้ใช้งานสร้างแอคเคาท์ขึ้นมาเพื่อให้แอพฯสามารถส่ง Token ไปให้ผู้ใช้งานได้ โดยใส่ Email และ Password ลงไปและกด “Sign Up”

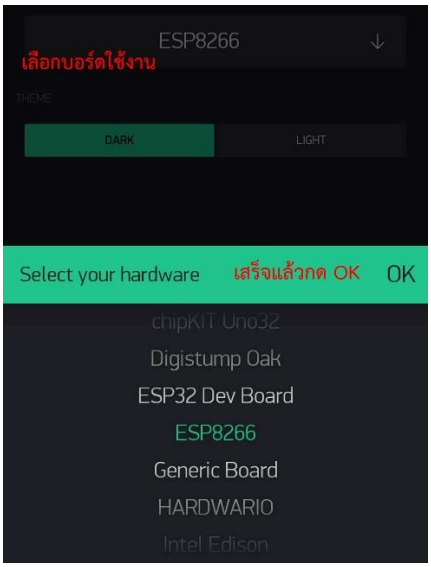


7.เมื่อทำการสร้างแอคเคานท์เสร็จแล้วก็จะปรากฏหน้าจอตามรูปข้างล่างขึ้นมา และให้เลือกที่เมนู “New Project”

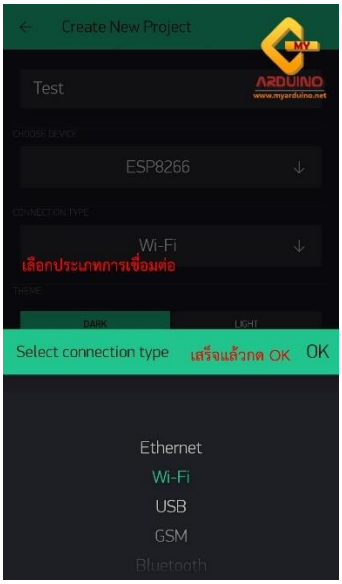
8.เมื่อเข้ามาแล้วให้ทำการตั้งชื่อโปรเจคที่ต้องการให้ช่อง Project Name



9.ในเมนูที่ 2 ให้เลือกประเภทของบอร์ดที่ผู้ใช้งานต้องการเชื่อมต่อในที่นี้ทางเราใช้ ESP8266

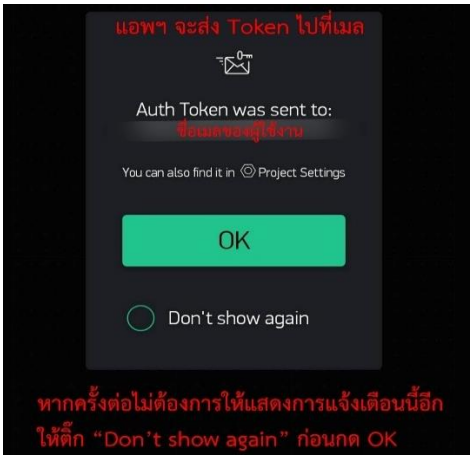


10.ในเมนูที่ 3 ให้เลือกประเภทของการเชื่อมต่อ (ในที่นี้ทางเราใช้ Wi-Fi)

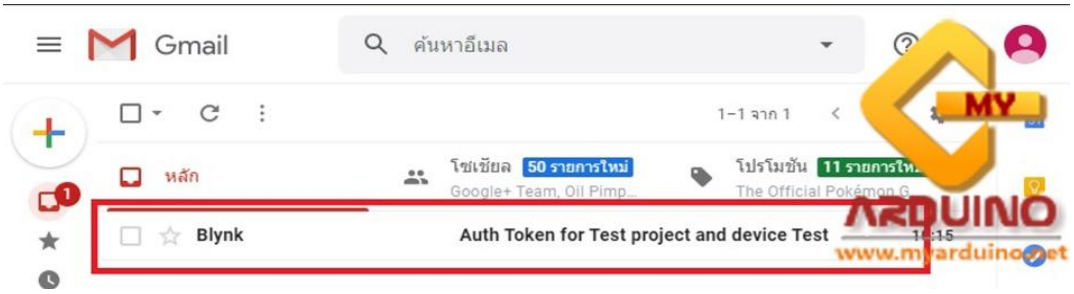


11.ผู้ใช้งานสามารถเลือกธีมการใช้งานได้ โดยจะมี 2 ธีมคือ ธีมมืด และธีมสว่าง จากนั้นก็ให้กด “Create”

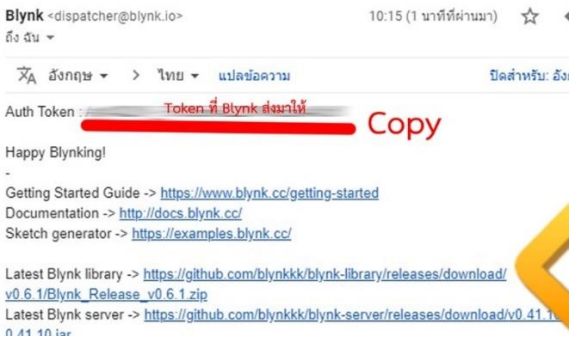
12.เมื่อทำการกด “Create” แล้วจะปรากฏหน้าต่างแจ้งเตือนขึ้นมาว่าทางแอพฯได้ส่ง Token ไปที่ Email ของผู้ใช้งานแล้ว จากนั้นก็กด OK (ในครั้งต่อไปหากผู้ใช้งานไม่ต้องการให้แสดงหน้าต่างการแจ้งเตือนนี้ขึ้นมาอีกให้เลือก “Don’t show again” ก่อนกด OK)



13.ให้เข้าไปที่ Email ที่ผู้ใช้งานทำการสร้างแอคเคานท์กับ Blynk ไว้ และจะพบกับ Email ของ Blynk

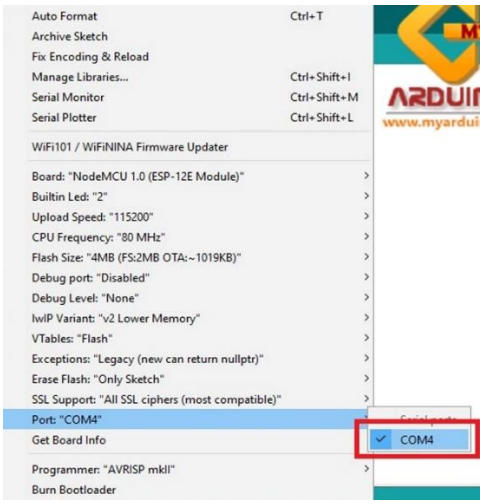


14.เมื่อกดเข้ามาในหน้าต่างเนื้อหาเมลแล้ว ก็จะพบกับ Auth Token ให้ผู้ใช้งานทำการ Copy Token ตามเส้นแดงในรูปข้างล่าง

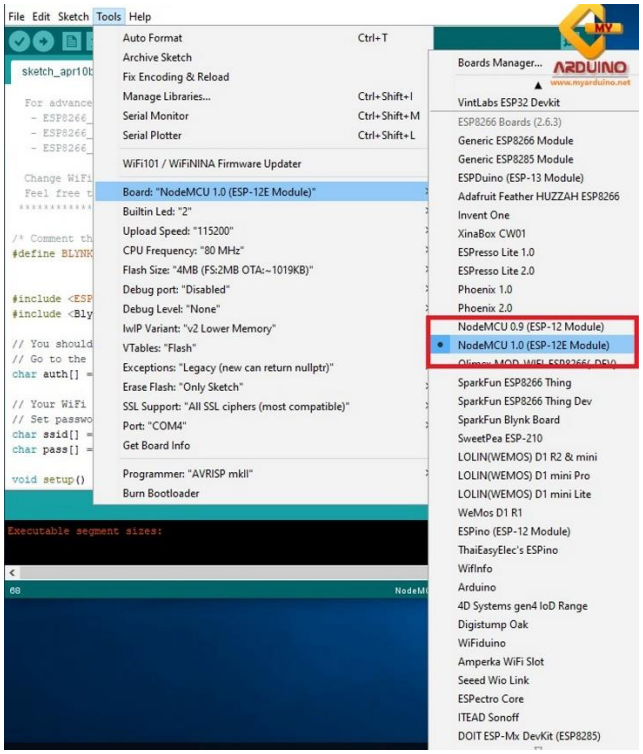


15.นำ Token ที่ Copy ไว้วางทับลงไปในช่องโค๊ดตามเส้นสีแดงในรูปข้างล่าง

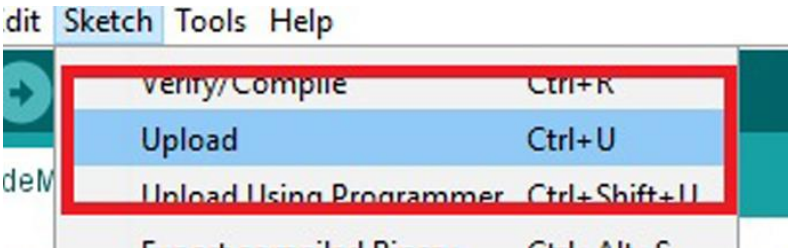
16.จากนั้นทำการเลือก Port ที่เชื่อมต่อกับบอร์ด



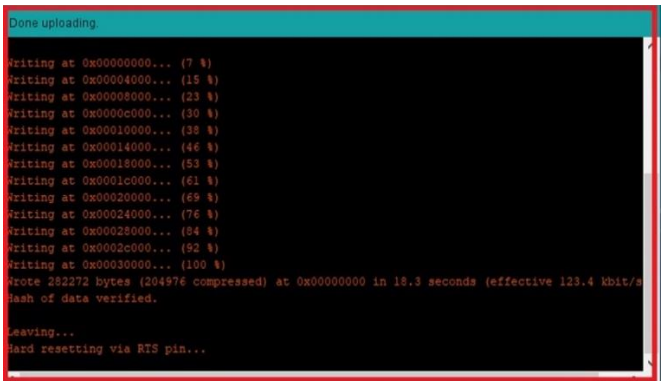
17.เลือกประเภทของบอร์ดที่ผู้ใช้งานต้องการ



18.กด Upload เพื่ออัพโหลดโค๊ดลงบอร์ด

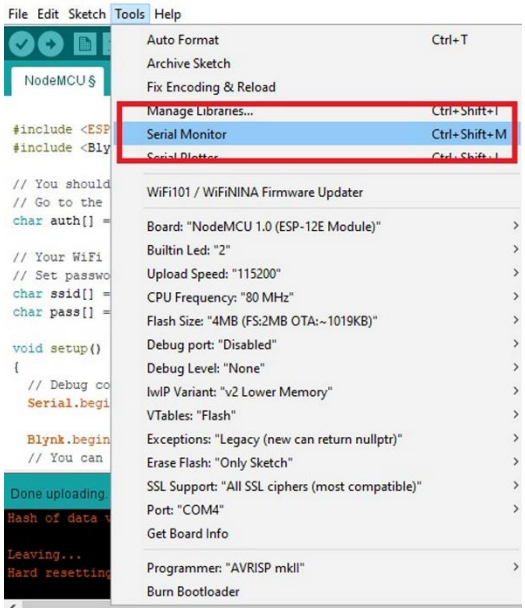


19.เมื่อทำการอัพโหลดโค๊ดลงบอร์ดเสร็จแล้วจะมีลักษณะตามรูปข้างล่าง

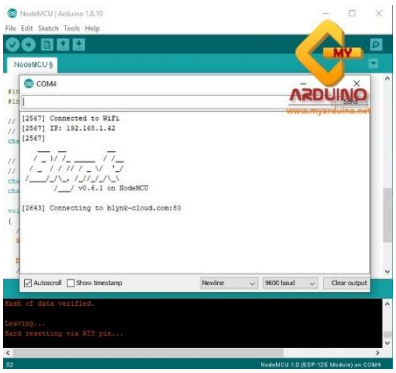


20.ในขณะเดียวกันในหน้าจอมือถือก็จะมีสถานะแจ้งเตือนขึ้นมาว่าโปรเจคของผู้ใช้งานได้ทำการเชื่อมต่อกับอุปกรณ์แล้ว (ในที่นี้ทางเราตั้งชื่อโปรเจคว่า Test)

21.จากนั้นให้ทำการเปิดหน้าต่าง Serial Monitor ขึ้นมา



22.หากทุกอย่างถูกต้องในหน้าต่าง Serial Monitor จะแสดงข้อความตามรูปข้างล่าง



ประวัติผู้จัดทำโครงงาน

นรจ.วัชรพล บุญพินิจ เหล่า อิเล็กทรอนิกส์
ที่อยู่ 200 ม.1 ต.นาแก อ.นาแก จ.นครพนม 48130
จบจาก โรงเรียนนาแกสามัคคีวิทยา จังหวัด นครพนม

นรจ.ภานุเดช อาบทอง เหล่า อิเล็กทรอนิกส์
ที่อยู่ 23/4 ม.6 ต.ตรมไพร อ.ศรีขรภูมิ จ.สุรินทร์ 32110
จบจาก โรงเรียนศรีขรภูมิพิสัย จังหวัด สุรินทร์

นรจ.ภานุพงศ์ ไชยสลี เหล่า อิเล็กทรอนิกส์
ที่อยู่ 18/60 ต.ศาลายา อ.พุทธมณฑล จ.นครปฐม 73170
จบจาก โรงเรียนมัธยมวัดมกุฏกษัตริย์ จังหวัด กรุงเทพฯ

นรจ.รัฐศาสตร์ กัลโยธิน เหล่า อิเล็กทรอนิกส์
ที่อยู่ 38/28 ต.นางรอง อ.นางรอง จ.บุรีรัมย์ 31110
จบจาก โรงเรียนนางรองพิทยาคม จังหวัด บุรีรัมย์

ปัญหา

1.ในกรณีที่บอร์ด ESP8266 ไม่มีไฟเลี้ยงจะทำให้ไม่สามารถสั่งการทำงานของระบบได้

2.บอร์ด ESP8266 จำเป็นที่จะเชื่อมต่ออินเทอร์เน็ตตลอดเวลาเพื่อที่จะสั่งการปรับความเร็วใบพัด

3. ปัญหาการเลือกอุปกรณ์การขาดความเข้าใจในอุปกรณ์ที่จะนำมาใช้

4..บอร์ด ESP 8266 ไม่ทำงานเนื่องจากลงโค้ดไม่ตรงตามที่ต้องการ

5. ในระหว่างที่ใช้โหมดออโต้แล้วเปลี่ยนไปใช้โหมดอื่นนั้นหลังจากที่เราจะกลับมาใช้โหมดออโต้อีกครั้งพบว่าเซ็นเซอร์จะหยุดการทำงาน

ข้อเสนอแนะ

1. การเลือกวัสดุอุปกรณทาง รร.อล.กวก.อล.ทร. ควรจะเปิดโอกาสใหนักเรียนไดศึกษาทดสอบอุปกรณที่ต้องใชในการจัดทำโครงงาน รวมถึงทฤษฎีการทำงาน วิธีการติดตั้ง รวมทั้งขนาดที่เหมาะสม เพื่อที่จะเลือกใช้อุปกรณ์ได้อย่างถูกต้อง

2. โปรแกรม Arduino นั้นมีความซับซ้อน ผู้ศึกษาต้องใช้ความมุ่งมั่น และความพยายาม เพื่อทำความเข้าใจอย่างลึกซึ้งในการศึกษาโปรแกรม รวมไปถึงต้องทุ่มเทให้เวลาในการศึกษาโปรแกรม

จัดทำโดย

นรจ.วัชรพล บุญพินิจ

นรจ.ภานุพงศ์ ไชยสลี

นรจ.ภานุเดช อาบทอง

นรจ.รัฐศาสตร์ กัลโยธิน

พัดลมปรับความเร็วตามอุณหภูมิ

(Adjustable Speed Fan by Temperature)

โครงงานนี้เป็นโครงงานพัดลมปรับความเร็วตามอุณหภูมิโดยใช้หลักการ Sensor ตรวจจับค่าอุณหภูมิและค่าความชื้นสัมพัทธ์ แล้วประมวลผลเพื่อทำการปรับความเร็วของใบพัดของพัดลม มีส่วนประกอบหลักคือ Relay Board, Adapter DC 12 V, พัดลมตั้งพื้น, Node MCU ESP8266 และ DHT 22

การทำงานมี 2 ระบบ คือระบบ Manual และระบบ Automatic โดยระบบ Manual จะทำการปรับความเร็วใบพัดผ่านทาง BLYNK Application เมื่อ BLYNK App มีปัญหาหรือสัญญาณ wireless ไม่เสถียร ก็จะเปลี่ยนไปใช้การทำงานแบบ Manual ได้ โดยการกดสวิตช์พัดลมได้ตามปกติ ค่าความชื้นที่บันทึกไว้ใน BLYNK App เมื่อเซ็นเซอร์ทำการตรวจวัดค่าความชื้น ค่าที่ได้จะนำไปประมวลผลเพื่อทำการปรับความเร็วใบพัดของพัดลม จากทดลองถ้าค่าอุณหภูมิมีค่าต่ำ พัดลมจะหมุนช้าลง และค่าอุณหภูมิสูง ใบพัดลมจะหมุนเร็วขึ้น

ที่มาและความสำคัญ

โดยปกติแล้วพัดลมที่มีการใช้งานอยู่กับในปัจจุบันจะเป็นแบบปกติ และแม้ว่าการใช้พัดลมแทนเครื่องปรับอากาศจะช่วยลดการใช้พลังงานไฟฟ้าได้อย่างมาก แต่หากใช้อย่างไม่ประหยัดก็ยังถือว่าเป็นการสิ้นเปลืองพลังงานโดยเปล่าประโยชน์เช่นกัน โดยเฉพาะอย่างยิ่งการใช้พัดลมขณะนอนหลับพักผ่อนในเวลากลางคืน ซึ่งสภาพอากาศเปลี่ยนแปลงอยู่บ่อยๆและอุณหภูมิแต่ละองศาแตกต่างกันออกไปตามสภาพอากาศ ทำให้ร่างกายของเราปรับตัวตามสภาพอากาศไม่ทัน จึงทำให้ไม่สบายหรือเป็นหวัดอยู่บ่อยๆ ยิ่งสภาพอากาศตอนกลางคืนจะหนาวกว่ากลางวัน ซึ่งพัดลมที่เราใช้กันอยู่เวลาเรานอนหลับแล้วพัดลมยังทำงานอยู่ยิ่งดึกความเย็นยิ่งมากขึ้น แต่พัดลมที่เราเปิดเอาไว้มันไม่สามารถเปลี่ยนระดับความเร็วลงตามอุณหภูมิของห้องนอนหรือสภาพอากาศได้นั่นเอง

ประโยชน์ที่คาดว่าจะได้รับ

1.)สร้างชุดควบคุมพัดลมปรับความเร็วตามระดับอุณหภูมิ

2.)ได้ความรู้เกี่ยวกบัการประดิษฐ์พัดลมอัตโนมัติปรับระดับตามอุณหภูมิ

3.) เพิ่มความสะดวกสบายใช้การใช้งาน

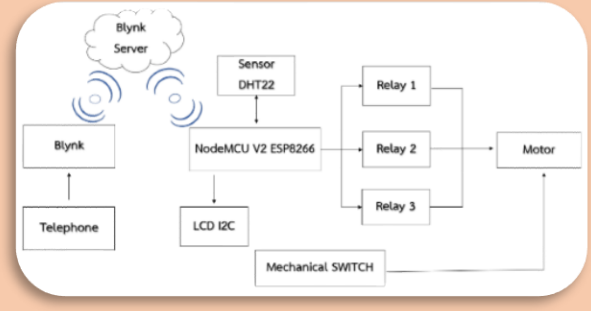
วัตถุประส่งโครงงาน

1.)เพื่อศึกษาขั้นตอนวิธีการควบคุมผ่านทางแอปพลิเคชั่น

2.)เพื่อทดสอบประสิทธิภาพของพัดลมอัตโนมัติปรับระดับตามอุณหภูมิ

3.)เพื่อนำความรู้ภาคทฤษฎีมาบูรณาการใช้ในการทำโครงงาน

หลักการทำงาน

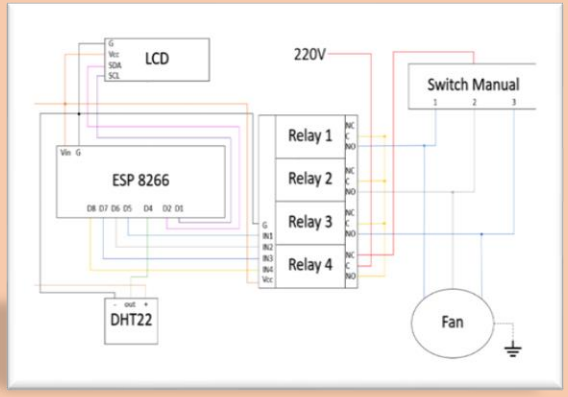


1. โหมดออโต้ เซ็นเซอร์จะคอยวัดอุณหภูมิเพื่อส่งข้อมูลมาที่บอร์ดเพื่อไปควบคุมการทำงานของพัดลมและไปแสดงค่าอุณหภูมิ ณ ขณะนั้นที่จอ LCD

2.โหมดไร้สาย โทรศัพท์จะทำหน้าที่คล้ายกันกับรีโมตควบคุมการทำงานของพัดลม โดยผ่านทาง

Blynk app จะส่งข้อมูลให้ที่บอร์ดเพื่อควบคุมการทำงานของพัดลมผ่านทาง wireless

3.โหมดปกติ สามารถมากดที่สวิตช์พัดลมได้ตามปกติ



การทำงานมี 2 ระบบ คือระบบ Manual และระบบ Automatic โดยระบบ Manualจะทำการเปิดและปิดพัดลมผ่านทางการกดสวิตช์ เมื่อเราต้องการเปลี่ยนสวิตช์พัดลมก็สามารถกดผ่านทาง Blynk App ได้ ส่วนระบบ Automatic จะทำงานโดยนำค่าอุณหภูมิที่บันทึกไว้ใน Blynk App เมื่อเซ็นเซอร์ทำการตรวจวัดค่าอุณหภูมิค่าที่ได้จะนำไปประมวลผลเพื่อทำการปรับความเร็วของมอเตอร์ทำงานพัดลมหมุนตาม จากทดลองถ้าอุณหภูมิต่ำกว่าค่าที่กำหนดพัดลมก็จะไม่ทำงาน หากอุณหภูมิมีค่าเท่ากับหรือมากกว่าที่กำหนดไว้พัดลมก็จะทำงาน

ผลการทดลอง

อุณหภูมิ (องศา)	ระดับพัดลม (เบอร์)	จำนวน(นาย)
ต่ำกว่า 25	0	3
25-26	1	2
26-27	2	2
มากกว่า 28	3	3

เมื่อเปิดพัดลมปรับความตามระดับอุณหภูมิ พัดลมจะปรับตามอุณหภูมิในห้อง จอLCD จะแสดงอุณหภูมิตามที่เซ็นเซอร์วัดได้ ถ้าอุณหภูมิในห้องต่ำกว่า 25 องศา จะหยุดทำงานอัตโนมัติ เมื่ออุณหภูมิเพิ่มขึ้นเป็น 25-26 องศาพัดลมจะทำงานที่เบอร์ 1 แต่ถ้าอุณหภูมิเพิ่มขึ้นเป็น 27-28 องศาพัดลมจะทำงานที่เบอร์ 2 และตั้งแต่ 28 องศา ขึ้นไปพัดลมจะทำงานที่เบอร์ 3