ระบบรดน้ำต้นไม้และดูค่าสถานะผ่านเน็ตพาย

Plant Watering System via NETPIE Application

จัดทำโดย

นรจ.ภานุวัฒน์	จานุวี
นรจ.ยศพัฒน์	คำแสน
นรจ.นัฐพงษ์	ไตรพันธุ์
นรจ.พันธกานต์	ไชยบุญทัน
นรจ.จตุพล	จุ้ยสุข
นรจ.จารุวิชย์	ทองชู

โครงงานนี้เป็นส่วนหนึ่งของการศึกษาหลักสูตรนักเรียนจ่าทหารเรือชั้นปีที่ ๒
พรรคพิเศษ เหล่า ช่างยุทธโยธา (อิเล็กทรอนิกส์) ปีการศึกษา ๒๕๖๔

โรงเรียนอิเล็กทรอนิกส์ กองวิทยาการ กรมอิเล็กทรอนิกส์ทหารเรือ

หัวข้อโครงงาน ระบบรดน้ำต้นไม้และดูค่าสถานะผ่านเน็ตพาย
(Plant Watering System via NETPIE Application)

ผู้จัดทำ	นรจ.ภานุวัฒน์	จานุวี	หัวหน้ากลุ่ม
	นรจ.ยศพัฒน์	คำแสน	รองหัวหน้ากลุ่ม
	นรจ.นัฐพงษ์	ไตรพันธุ์	สมาชิกกลุ่ม
	นรจ.พันธกานต์	ไชยบุญทัน	สมาชิกกลุ่ม
	นรจ.จตุพล	จุ้ยสุข	สมาชิกกลุ่ม
	นรจ.จารุวิชย์	ทองชู	สมาชิกกลุ่ม

ครูที่ปรึกษา	**ว่าที่** ร.ท.กฤษณะ	เพิ่มคำ
	ว่าที่ ร.ท.ฉลาม	แก้ววิเศษ
	พ.จ.อ.นัฐพล	ลิมบุรีธรรม

ปีการศึกษา ๒๕๖๔

สารบัญ

กิตติกรรมประกาศ ..... ก

บทคัดย่อ ..... ข

บทที่ 1 1

บทนำ 1

1.1 ที่มาและความสำคัญ 1

1.2 วัตถุประสงค์ 1

1.3 ขอบเขตการทำโครงงาน 1

1.4 ประโยชน์ที่ได้รับจากโครงงาน 1

บทที่ 2 2

ทฤษฎีและงานวิจัยที่เกี่ยวของ 2

2.1 เซ็นเซอร์วัดความชื้นในดิน (Soil Moisture Capacitive Sensor V2) 2

2.2 เซ็นเซอร์วัดความชื้นและอุณภมิ DHT 11 3

2.3Relay Module 2 ช่อง 5 โวลต์ 10A 250V 4

2.4 NETPIE 5

2.5 NodeMCU V3 3

2.6 ปั๊มน้ำ DC12V รุ่น AS180-12250A 8

2.7 Adapter dc 9Vdc 9

2.8 Switching Power supply 12Vdc 10

บทที่ 3 11

วิธีการดำเนินงาน 11

3.1 วิธีการดำเนินงาน 11

3.2 แผนการดำเนินงาน 14

3.3 วัสดุอุปกรณ์ 15

3.4 ขั้นตอนการดำเนินงาน 16

บทที่ 4 18

ผลการทดลอง 18

4.1 ขั้นตอนการทดสอบ 18

บทที่ 5 19

สรุป ปัญหาและข้อเสนอแนะ 19

บรรณานุกรม 20

ภาคผนวก 21

กิตติกรรมประกาศ

การจัดทำโครงงานสิ่งประดิษฐ์ในครั้งนี้สำเร็จได้โดยการค้นคว้าหาความรู้ การให้คำปรึกษา และการกำกับดูแลจาก ว่าที่ ร.ท.กฤษณะ เพิ่มคำ ว่าที่ ร.ท.ฉลาม แก้ววิเศษ และ พ.จ.อ.นัฐพล ลิมบุรีธรรม ครูที่ปรึกษาโครงงาน

ขอขอบพระคุณ คณะครูฝ่ายศึกษาที่ให้การสนับสนุนในด้านความรู้และคำแนะนำ รวมถึงสนับสนุนอุปกรณ์ที่ขาดเหลือเกี่ยวกับโครงงานสิ่งประดิษฐ์

ขอขอบพระคุณ คณะครูห้องวิทยาการที่ให้การสนับสนุนและคำแนะนำเรื่องการเขียนโปรแกรมควบคุมอุปกรณ์ที่ใช้ในการทำโครงงงานสิ่งประดิษฐ์

คุณความดีที่เกิดจากโครงงานสิ่งประดิษฐ์นี้ขอมอบให้กับคณะครู และอาจารย์ทุกท่านที่ให้การสนับสนุนและประสิทธิ์ประสานวิชาทำให้นักเรียนจ่ามีความรู้ ความเข้าใจ ส่งผลให้การทำโครงงานสิ่งประดิษฐ์ชิ้นนี้สำเร็จลุล่วงไปได้ด้วยดี

จัดทำโดย

นรจ.ภานุวัฒน์ จานุวี
นรจ.ยศพัฒน์ คำแสน
นรจ.นัฐพงษ์ ไตรพันธุ์
นรจ.พันธกานต์ ไชยบุญทัน
นรจ.จตุพล จุ้ยสุข
นรจ.จารุวิชย์ ทองชู

บทคัดย่อ

โครงงานนี้เป็นโครงงานระบบรดน้ำต้นไม้และดูค่าสถานะผ่านเน็ตพาย โดยใช้หลักการ Sensor ตรวจจับค่าความชื้นของดิน แล้วประมวลผลเพื่อทำการรดน้ำต้นไม้ มีส่วนประกอบหลักคือ Soil Moisture Capacitive Sensor V2, ปั๊มน้ำ DC12V, Node MCU ESP8266 และ DHT 11

การทำงานมี 2 ระบบ คือระบบ Manual และระบบ Automatic โดยระบบ Manual จะทำการเปิดและปิดปั๊มน้ำผ่านทาง NETPIE Application เมื่อ NETPIE App มีปัญหา ระบบ Automatic จะทำงานโดยนำค่าความชื้นที่บันทึกไว้ใน NETPIE App เมื่อเซ็นเซอร์ทำการตรวจวัดค่าความชื้น ค่าที่ได้จะนำไปประมวลผลเพื่อทำการรดน้ำต้นไม้ จากทดลองถ้าค่าความชื้นต่ำกว่าค่าที่กำหนดปั๊มน้ำก็จะทำงาน และค่าความชื้นมากกว่าค่าที่กำหนดปั๊มน้ำจะหยุดทำงาน

ว่าที่ ร.ท.

ครูที่ปรึกษาโครงงาน

บทที่ 1
บทนำ

1.1 ที่มาและความสำคัญ

เนื่องด้วยปัจจุบันสภาพอากาศค่อนข้างมีความแปรปรวนและยากที่จะควบคุมส่งผลกระทบต่อการดำเนินชีวิตของมนุษย์ สัตว์ สิ่งแวดล้อม รวมถึงพืช แต่ข้อจำกัดของพืชคือไม่สามารถเลือกสภาพแวดล้อมที่เหมาะสมในการเจริญเติบโตได้ ทำให้เกิดแนวคิดในการนำ IOT มาใช้เพื่อให้เกิดความสะดวกสบายในการใช้งานและควบคุมผลผลิตได้ เนื่องจาก IOT สามารถทำประโยชน์ได้หลายอย่าง เช่น การแจ้งสถานะค่าความชื้น อุณหภูมิการควบคุม อุปกรณ์ไฟฟ้าต่างๆ จากระยะไกลได้ และ NETPIE (Network Platform for Internet of Everything) ก็เป็นหนึ่งใน Platform ที่ให้บริการเชื่อมต่อการสื่อสาร IOT สามารถนำมาประยุกต์ใช้ในการสื่อสารติดตาม ควบคุม โดยไม่ต้องกังวลเรื่องการติดตั้งดูแลระบบสื่อสาร หรือฐานข้อมูลใดๆ ในขณะเดียวกันก็ยังช่วยนักพัฒนาเข้าถึงอุปกรณ์อิเล็กทรอนิกส์ต่างๆ ได้ง่ายขึ้น โดย NETPIE นั้นเป็น Platform สัญชาติไทยที่พัฒนาขึ้นโดยศูนย์เทคโนโลยี อิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ (NECTEC) และสำนักงานพัฒนาวิทยาศาสตร์และเทคโนโลยีแห่งชาติ (สวทช.)

1.2 วัตถุประสงค์

1.2.1 เพื่อสร้างระบบการรดน้ำต้นไม้ ผ่าน NETPIE APP สามารถติดตามผลและควบคุมการทำงานได้

1.2.2 เพื่ออำนวยความสะดวกในการรดน้ำต้นไม้

1.2.3 เพื่อนำความรู้ภาคทฤษฎีมาบูรณาการใช้ในการทำโครงงาน

1.2.4 เพื่อพัฒนาผลงานสิ่งประดิษฐ์ไปใช้ในชีวิตประจำวันได้จริง

1.3 ขอบเขตการทำโครงงาน

1.3.1 สามารถเชื่อมต่อกับสมาร์ทโฟน สั่งการผ่านสมาร์ทโฟนได้

1.3.2 สร้างแบบจำลองการรดน้ำอัตโนมัติใช้ภายในพื้นที่สวนแปลงสาธิต

1.3.3 สามารถดูสถานะอุณหภูมิ ความชื้นบริเวณแปลงสาธิต

1.3.4 ระบบรดน้ำอัตโนมัติกรณีที่ไม่สามารถเชื่อมต่อ NETPIE ได้

1.3.5 ติดตั้งในพื้นที่ที่มีสัญญาณ Internet

1.4 ประโยชน์ที่ได้รับจากโครงงาน

1.4.1 ได้ระบบรดน้ำต้นไม้ผ่าน NETPIE ที่สามารถติดตามผลและควบคุมการทำงานได้

1.4.2 เป็นต้นแบบที่จะสามารถให้หน่วยงานอื่นๆนำไปใช้งานและต่อยอดได้

1.4.3 ได้ความรู้ที่ได้จากภาคทฤษฎีมาใช้ในการทำโครงงาน

1.4.4 ได้ผลงานสิ่งประดิษฐ์ไปใช้ในชีวิตประจำวัน

1.4.5 ช่วยอำนวยความสะดวกในการรดน้ำต้นไม้

1.4.6 ช่วยเพิ่มประสิทธิภาพในด้านอุตสาหกรรมการเกษตร

บทที่ 2
ทฤษฎีที่เกี่ยวข้อง

2.1 เซ็นเซอร์วัดความชื้นในดิน (Soil Moisture Capacitive Sensor V2)

วัดความชื้นในดิน หรือใช่เป็นเซ็นเซอรน้ำ สามารถตอใชงานกับไมโครคอนโทรลเลอรโดยใช อนาล็อกอินพุตอานคาความชื้น หรือเลือกใช่สัญญาณดิจิตอลที่สงมาจากโมดูล สามารถปรับความไวได้ด่วยการปรับความต้านทานแบบปรับคาได้ (Trim pot)

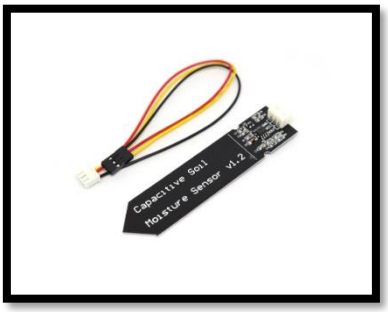
หลักการทำงาน

เซ็นเซอร์วัดความชื้นในดิน Capacitive Soil Moisture Sensor V2 เป็นเซ็นเซอร์วัดความชื้นในดินแบบ Capacitive ซึ่งปรับปรุงวิธีการวัดจากแบบเดิมที่เป็นแบบ resistor ใช้แผ่นทองแดงไปสัมผัสกับดินโดยตรง ซึ่งมีข้อเสียคือหัววัดจะไม่ทน เป็นสนิม และชำรุดง่าย ทำให้เกิดค่าคลาดเคลื่อนได้ง่าย เซ็นเซอร์วัดความชื้นในดินแบบ Capacitive ใช้หลักการตรวจสอบประจุของวัสดุ ถ้ามีค่าประจุมากแสดงว่ามีความชื้นมาก แผ่นเซ็นเซอร์จึงไม่จำเป็นต้องสัมผัสกับดินหรือวัสดุที่ต้องการวัดโดยตรง จึงทนทานและแม่นยำกว่า

โมดูลวัดความชื้นในดินนี้ให้ค่าเอาต์พุตเป็นแบบ Analog 0 - 3 VDC จึงเหมาะกับไมโครคอนโทรลเลอร์ MCU รุ่นต่างๆ เช่น Arduino Raspberry pi เป็นต้น

คุณสมบัติ

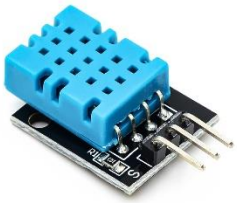
- วัดค่าความชื้นในดินโดยอาศัยหลักการของประจุไฟฟ้า
- ขั้ววัดมีฉนวนจึงไม่สัมผัสกับความชื้นในดินโดยตรง ทำให้ไม่มีปัญหาขั้ววัดเสื่อมสภาพ
- ใช้แรงดันไฟฟ้าเลี้ยง 3.3 - 5.5V
- แรงดันเอาต์พุตอยู่ในช่วง 0 - 3V
- ช่องเสียบ Interface แบบ PH 2.0 - 3P



รูปที่ 2.1 Capacitive Soil Moisture Sensor V2

ที่มา: https://images.app.goo.gl/uwwDKef3ra422S5N9

2.2 เซ็นเซอร์วัดอุณหภูมิและความชื้น DHT 11



รูปที่ 2.2 Sensor วัดอุณหภูมิ และความชื้น DHT 11

ที่มา: https://images.app.goo.gl/CPjKT6SCWQiDrYoP6

หลักการทำงาน

DHT 11 เป็น Sensor เพื่อการรับรู้หรือตรวจวัดระดับอุณหภูมิและวัดความชื้นสัมพัทธ์เบอร์ DHT 11 การเชื่อมต่อเป็นแบบหนึ่งสายสัญญาณ นั่นคือใช้พอร์ตของไมโครคอนโทรลเลอร์ หรือ GPIO ของ Raspberry Pi เพียงหนึ่งพอร์ตในการทำงาน

คุณสมบัติ

- ย่านวัดความชื่น 20 - 90% RH ค่าความแม่นยำ +- 5% RH
- ย่านวัดอุณหภูมิ 0 - 50 องศา ค่าความแม่นยำ +- 2 องศาเซลเซียส
- มี 3 ขา
- ใช้ไฟเลี้ยง 3 - 5.5 VDC
- อ่านค่าสัญญาณทุก 1 วินาที

ตำแหน่งขาของ DHT 11

- ขาที่ 1 VCC ใช้ไฟเลี้ยง 3 – 5.5 V
- ขาที่ 2 DATA เป็นขาสัญญานข้อมูล
- ขาที่ 3 GND กราวด์

รูปที่2.3 ตำแหน่งขาของ Sensor วัดอุณหภูมิ และความชื้น DHT 11

ที่มา: https://images.app.goo.gl/9utGCjaVK8HeqUaY6

2.3 รีเลยโมดูล (Relay Module)

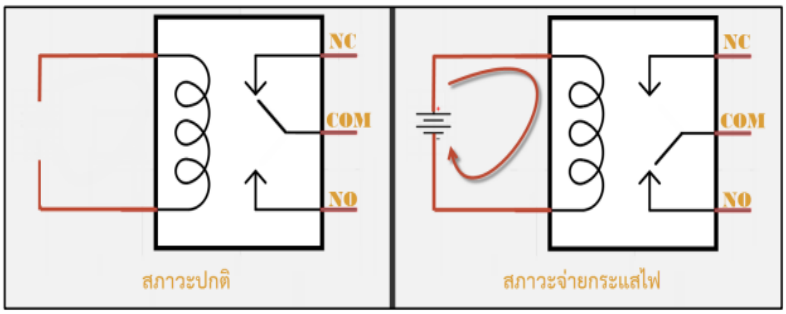
รูปที่ 2.4 รีเลยบอรด (Relay Board)

ที่มา: https://images.app.goo.gl/E1tB3FbLT4L7fEXR9

เปนรีเลยบอรดชนิด 2 ชอง Output ต้องการแหล่งจ่ายไฟ 5 Vdc สงสัญญาณควบคุมแบบ Active Low ซึ่งสั่งการได้งายถาตองการใหรีเลยทำงานสงสัญญาณ 0 ออกไป และถ้าต่องการใหรีเลย์หยุดทำงานสงสัญญาณ1 ออกไป วงจรเปนแบบแยกกราวด Op-to isolated relay ปลอดภัยตอวงจรไมโครคอนโทรเลอร์

คุณสมบัติ

- การเชื่อมตอ Arduino Relay module มาตรฐานที่สามารถใชควบคุมไดโดยตรงจากไมโครคอนโทรนเลอร เชน Arduino , 8051, AVR, PIC, DSP, ARM, ARM, MSP430, TTL logic
- ใชแรงดัน 5 Vdc
- Relay Module สามารถใชแรงดัน 5 Vdc จากบอรด Arduino ที่มีขา 5 Vdc ได
- ใชควบคุมไฟฟาแรงสูงไดที่ DC30V 10A , AC 250V 10A
- มีไฟบอกสถานะการทำงานของ Arduino Relay รีเลยทุกชอง
- เชื่อมตอดวยขั้วสกรู ทำใหติดตั้งไดงายและสะดวก
- ใชกระแสขับ relay แตละตัวที่ 15 - 20 mA
- การสงสัญญาณควบคุมรีเลยเปนแบบ Active low
- วงจรขับรีเลยเปนแบบแยกกราวด Op-to isolated Relay ปลอดภัยตอวงจร ไมโครคอนโทรเลอร Arduino Relay



รูปที่ 2.5 แสดงสภาวะการทำงานของรีเลย์

ที่มา: https://images.app.goo.gl/Btf9XNGHzd6BbZej

จุดตอใชงานมาตรฐาน

จุดตอ NC ยอมาจาก normal closed หมายความวาปิดปกติ หรือหากยังไมจายไฟให ขดลวดเหนี่ยวนำหนาสัมผัสจะต่อถึงกัน โดยทั่วไปเรามักตอจุดนี้เขากับอุปกรณหรือเครื่องใชไฟฟาที่ตองการให ทำงานตลอดเวลา

จุดตอ NO ยอมาจาก normal opened หมายความวาปกติเปิด หรือหากยังไม่จ่ายไฟให ขดลวดเหนี่ยวนำหนาสัมผัสจะไมต่อถึงกัน โดยทั่วไปเรามักตอจุดนี้เขากับอุปกรณหรือเครื่องใชไฟฟาที่ตองการควบคุมการเปิดปด เช่น โคมไฟสนามหนาบ้าน

จุดตอ COM ยอมาจาก common คือจุดร่วมที่ตอมาจากแหล่งจ่ายไฟ

2.4 NETPIE หรือ Network Platform for Internet of Everything

NETPIE หรือ Network Platform for Internet of Everything เป็นแพลตฟอร์มของการให้บริการ cloud เพื่อการสื่อสารกันระหว่างอุปกรณ์ IOT ต่างๆบนเครือข่ายอินเทอร์เน็ตซึ่งถูกพัฒนาขึ้นมาโดยศูนย์อิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติหรือ NECTEC เมื่อปี ค.ศ.2015 โดยมี NECTEC Server ทำหน้าที่เป็น Cloud Server หรือเป็นตัวกลางในการเชื่อมต่อระหว่างอุปกรณ์ IOT ที่มีอยู่อย่างมากมายบนเครือข่ายไร้สายเข้าด้วยกัน เพื่อให้สามารถติดต่อสื่อสารและรับส่งข้อมูลระหว่างกันได้ในแบบ Real – Time หรือสั่งการควบคุมไปยังอุปกรณ์ IOT ที่อยู่ห่างไกลตามสถานที่ต่างๆได้เท่าที่สัญญาณเครือข่ายและอินเทอร์เน็ตจะครอบคลุมทั่วถึง (เป็นเว็บไซต์ที่ไม่มีค่าใช้จ่าย)

คุณสมบัติหรือหน้าที่ของ NETPLE

- **Monitoring:** การแสดงค่าข้อมูลจากเซ็นเซอร์หรืออุปกรณ์ต่างๆ ในแบบ Real-Time ช่วยให้ผู้ใช้สามารถเฝ้าสังเกตและติดตามการทำงานได้อย่างต่อเนื่องตลอดเวลา
- **Controlling:** การควบคุมการทำงานของอุปกรณ์ต่างๆ ผ่านระบบ Cloud โดยผู้ใช้งาน
- **Data Storage:** การเก็บค่าข้อมูลที่อ่านได้จากเซ็นเซอร์หรืออุปกรณ์ต่างๆ
- **Notification:** การตรวจจับความผิดปกติในการทำงานของอุปกรณ์ และแจ้งเตือนมายังผู้ใช้งาน
- **Dashboard for monitor & control:** การแสดงผลและควบคุมการทำงานของอุปกรณ์ผ่านแผงควบคุมหรือ Dashboard

2.5 บอรดไมโครคอนโทรลเลอร NodeMCU ESP8266

NodeMCU เป็นบอร์ดที่ใช้ ESP8266 เป็น CPU สำหรับประมวลผลโปรแกรมต่างๆ มีข้อดีกว่า Arduino ตรงที่ตัวมันมีขนาดเล็กกว่า มีพื้นที่เขียนโปรแกรมลงไปมากกว่า และสามารถเชื่อมต่อกับ WiFi ได้บน บอร์ดรุ่นนี้ใช้ ESP8266 12E มีพื้นที่หน่วยความจำรอมสูงถึง 4MB เพียงพอสำหรับการเขียนโปรแกรมขนาดใหญ่ อีกทั้งภายในยังเป็น ARM ขนาดย่อมๆ ใช้ความถี่สูงถึง 40MHz ทำให้สามารถประมวลผลโค้ดโปรแกรมได้อย่างรวดเร็วเหมาะมากสำหรับงาน Smart Home และ IOT

ESP8266 เปนชื่อเรียกของชิฟและของโมดูล ESP8266 สำหรับติดตอสื่อสารบนมาตรฐาน WiFi ทำงานที่แรงดันไฟฟา 3.0 - 3.6V ทำงานใชกระแสโดยเฉลี่ย 80mA รองรับคำสั่ง deep sleep ในการประหยัดพลังงาน โดยใชกระแสนอยกวา 10 ไมโครแอมป สามารถ wake up กลับมาสงขอมูลใชเวลานอยกวา 2 มิลลิวินาที ภายในมี Low power MCU 32bit ทำใหเราเขียนโปรแกรมสั่งงานได มีวงจร analog to digital converter ทำใหสามารถอานคาจาก analog ไดความละเอียด 10 bit ทำงานไดที่อุณหภูมิ -40 ถึง 125 องศาเซลเซียส เมื่อนำชิฟ ESP8266 มาผลิตเปนโมดูลหลายรุน ก็จะขึ้นตนดวย ESP8266 แลวตามดวยรุน เชน ESP-01 ,ESP-03 , ESP-07 , ESP-12E ESP8266 ติดตอกับ WI-FI แบบ Serial สามารถเขียนโปรแกรมลงไปในชิฟโดยใช Arduino IDE ไดทำใหการเขียนโปรแกรมและใชงานเปนเรื่องงายใกล้เคียงกับการใช Arduino สามารถติดตออุปกรณอื่นๆ เซ็นเซอรตางๆ ในแบบของ Arduino ถามีพื้นฐาน Arduino อยูแลวก็จะเขาใจและใชงานไดรวดเร็ว โมดูล ESP8266 มีหลายรุนและมีรุนใหมพัฒนาออกมาเรื่อยๆ โดยโครงสราง และขาที่ใชงานก็จะมีลักษณะใกล้เคียงกัน คือ

- GPIO0 เปนขาสำหรับเลือกโหมด โดยเมื่อตอกับ GND จะเขาโหมดโปรแกรม เมื่อตองการใหทำงานปกติก็ไมตองทำการต่อ
- GPIO15 เปนขาที่ตองตอลง GND เพื่อใหโมดูลทำงาน
- CH_PD หรือ EN เปนขาที่ตองตอไฟ VCC เพื่อ pull up สัญญาณ ใหโมดูลทำงาน โมดูลบางรุนไมมีขา Reset มาให เมื่อตองการรีเซต ใหตอขา CH_PD กับ GND
- Reset ตอกับไฟ VCC เพื่อ pull up สัญญาณ โดยเมื่อตองการรีเซต ใหตอกับไฟ GND
- VCC เปนขาสำหรับจายไฟเลี้ยง ใชไฟเลี้ยง 3.0 - 3.6V
- GND ตอกับไฟ 0V
- GPIO เปนขาดิจิตอล INPUT/OUTPUT ทำงานที่ไฟ 3.3V
- ADC เปนขา Analog INPUT รับแรงดันสูงสุด 1V ความละเอียด 10bit หรือ 1024 คาเวลาโปรแกรมเพียงมองหาขาเหลานี้ แลวตอใหครบเทาที่มีขาใหตอ ก็สามารถใช้ได้กับโปรแกรม ESP8266 ไดทุกรุน

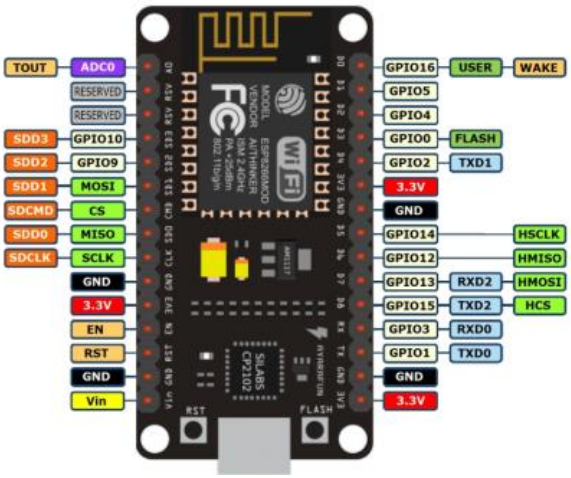
การเขียนโปรแกรม ESP8266 Arduino IDE

สำหรับการตอวงจรของ ESP8266 เพื่อเขียนโปรแกรมด่วย Arduino เขียนโค่ดเหมือนกันทุกรุน โดยแตละรุนจะมีขาไมเทากัน ดังนั้นเราจึงต่องเปรียบเทียบขา GPIO ให่ถูกต่องในการสั่งงานก็ใชไดแล่ว การเขียนโปรแกรมอัพโหลดโค่ดลงบอรด ESP8266 เกือบทุกรุน จะผานทาง Serial ที่ขา rx,tx โดยใชโมดูล USB TTL ซึ่งต่องเสียเวลาในการตอวงจรเพื่ออัพโหลดโค่ด อีกทั้งโมดูล ESP8266 หลาย ๆ รุนมีการตอขาที่เปนแบบเซอรเฟสเมาส ทำให่ไมสะดวกกับการตอทดลองบนบอรดทดลอง ดังนั้นจึงมีการรวมโมดูล USB TTL และตอวงจรขยายขา ESP8266 ให่เปนขาระยะหางขนาด 2.54 mm ซึ่งสามารถเสียบลงบอรดทดลอง ไดพอดีกลายเปนบอรด ESP8266 โดยหนึ่งในบอรดที่นิยมใช่งานคือ NodeMCU ซึ่งใชโมดูล ESP8266 ESP - 12E

บอรด ESP8266 NodeMCU V3

รูปที่ 2.6 NodeMCU Version3

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



รูปที่ 2.7 ตำแหนงขาตางๆ ของบอรด NodeMCU V3

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

NodeMCU V3 เปน ESP8266 - 12E รวมกับ USB TTL ที่ใชชิฟ CP2102 และขยายขาใหสามารถตอทดลองไดงายขึ้น มีปุม reset และ flash สำหรับใชโปรแกรม โดยใช Arduino IDE หรือโปรแกรมอื่นๆ ไดอยางสะดวก

2.6 ปมน้ำ DC12V รุน AS180-12250A

รูปที่ 2.8 ปมน้ำขนาดเล็กใชไฟตรงขนาด 12 Vdc

ที่มา: https://images.app.goo.gl/LzJfmSnajJeuLuzE8

ขอมูลอุปกรณ

- SIZE: กวาง X ยาว X สูง = 4.4 X 3.8 X 3.7 CM.
- แรงดันไฟฟา 12 Vdc
- กำลังไฟฟา 4.2 วัตต
- กินกระแส 0.35 A
- สายไฟยาว 2 เมตร
- สามารถใชงานกับแบตเตอรี่ 12V หรือ ใชกับแผงโซลาเซลลไมต่ำกวา 5 W
- อัตราการไหลสุงสุด : 280 ลิตร/ชม.
- อัตราการไหลสุงสุด : 4.7 ลิตร/นาที
- ระดับน้ำสูงสุด : 200 cm.

2.7 อุปกรณจายไฟแรงดัน 9 โวลต์ (Adapter 9V)

แหล่งจ่ายไฟ 9V 2A ให้อุปกรณ์อิเล็กทรอนิกส์ เสียบเขาบอรด Arduino ไดโดยตรง

รูปที่ 2.9 อุปกรณจายไฟแรงดัน 9 โวลต์

ที่มา: https://images.app.goo.gl/uu7VNKztqLX5F2717

รายละเอียด

- 1x AC 100-240V to DC 9V 2A Switching Power supply Converter Adaptor
- อะแด็ปเตอร์แบบสวิตซิ่ง จาก AC 100 - 240V เป็น ดีซี 9V 2 A
- ด้านปลายเป็นดีซีแจ๊คขนาด 5.5 x 2.5mm และใช้ได้กับ 5.5 x 2.1mm
- ขั้วในบวก ขั้วนอกลบ

2.8 Switching Power supply 12Vdc

ใช้แปลงไฟ 220 Vac ให้เป็นไฟ 12 Vdc จ่ายไฟให้กับปั๊มน้ำ

รูปที่ 2.10 Switching Power supply 12 Vdc

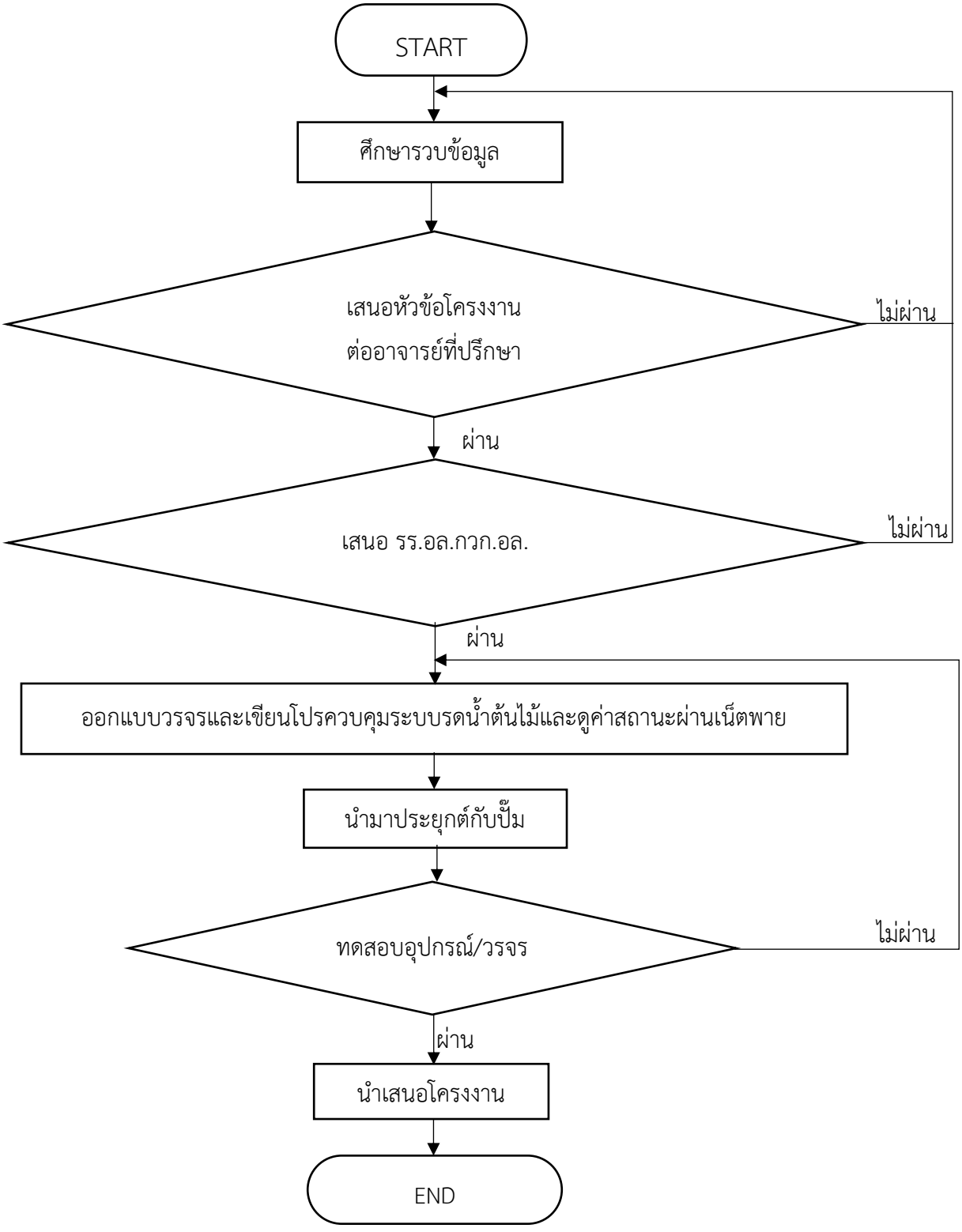
ที่มา: https://images.app.goo.gl/y14ZuJJp8aj6JA1Y7

รายละเอียด

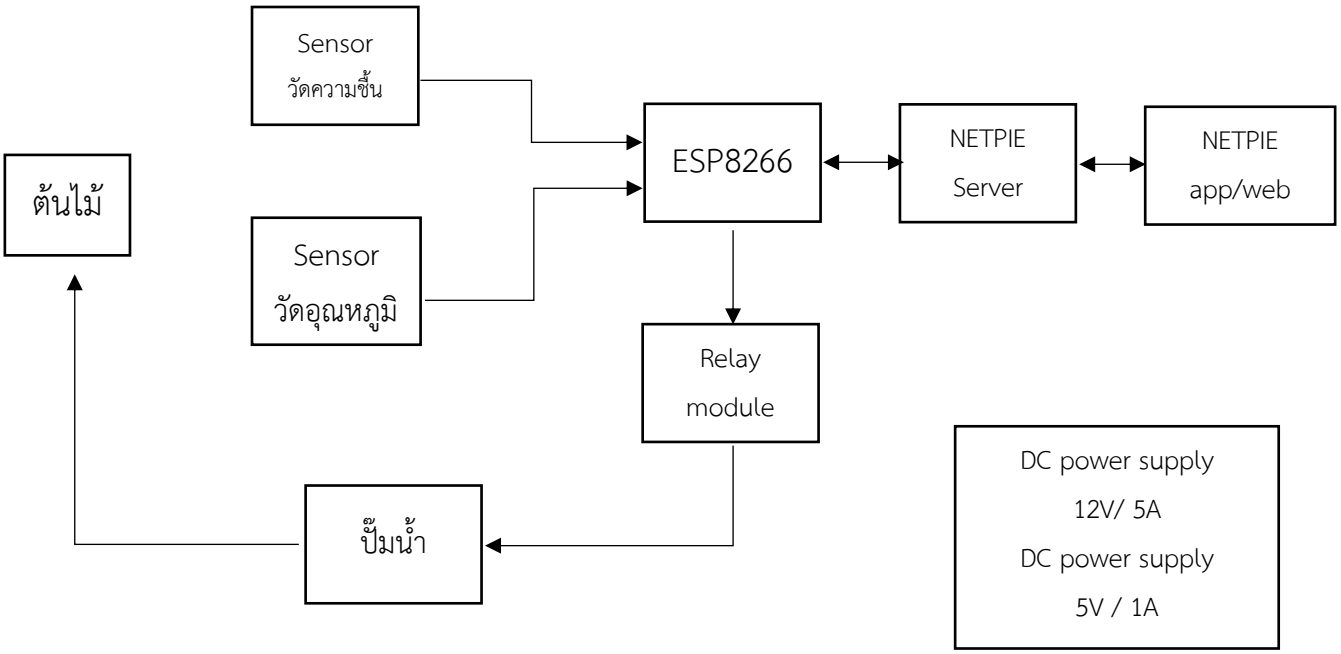
- แรงดันไฟฟ้า Input: 100 – 240 VAC 50/60Hz
- แรงดันไฟฟ้า Output : 12V DC 5A
- วัสดุที่ใช้ผลิต : Material Metal, Electronic Parts
- การป้องกัน: การป้องกันไฟขาด, ป้องกันไฟเกิน, ป้องกันแรงดันไฟฟ้า, ไฟกระชาก
- ขนาดสินค้า: 110 x 80 x 37cm
- น้ำหนัก : 200 g

บทที่ 3
การดำเนินงาน

3.1 วิธีการดำเนินงาน

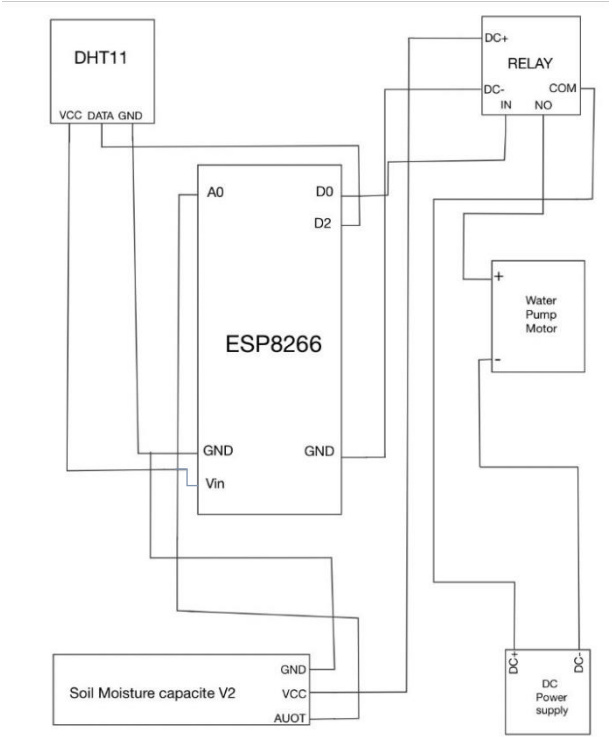


รูปที่ 3.1 ผังการดำเนินงานโครงงาน

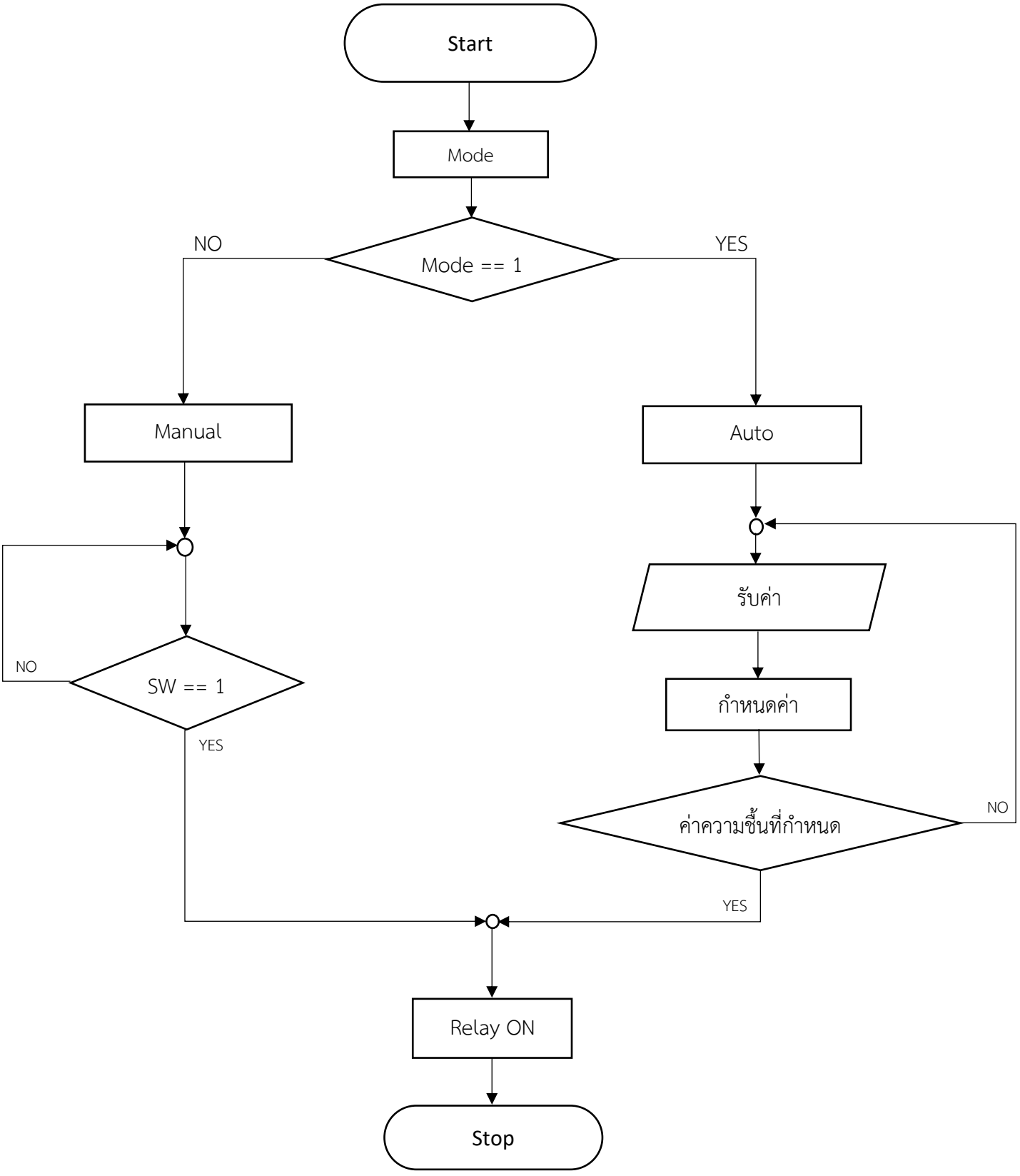


รูปที่ 3.2 บล็อกไดอะแกรม ระบบรดน้ำต้นไม้และดูค่าสถานะผ่านเน็ตพาย

บอร์ด ESP8266 จะรับค่า Analog จาก Sensors วัดความชื้นและส่งข้อมูลที่ได้ไปแสดงผลบน NETPIE App และสั่งการผ่าน NETPIE Server เพื่อให้ Relay Module on switch โดยมี Switching 12 Vdc คอยช่วยจ่ายไฟให้ปั๊มน้ำทำงาน



รูปที่ 3.3 Wiring diagram



รูปที่ 3.4 แผนผังการทำงานของระบบรดน้ำต้นไม้และดูค่าสถานะผ่านเน็ตพาย

3.2 แผนงานในการดำเนินโครงการ

แผนงานโครงงานสิ่งประดิษฐ์

หลักสูตร นรจ. พรรค พศ. เหลา ยย.(อิเล็กทรอนิกส - ไฟฟ้า) ชั้นปที่ 2 ประจำปการศึกษา 2564

โรงเรียนอิเล็กทรอนิกส กองวิทยาการ กรมอิเล็กทรอนิกส์

ลำดับ	รายการปฏิบัติ	ต.ค. 64				พ.ย. 64				ธ.ค. 64				ม.ค. 65				ก.พ. 65				มี.ค.65			
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1	ศึกษารวบรวมข้อมูล																								
2	เสนอโครงงาน ต่ออาจารย์ที่ปรึกษา																								
3	ออกแบบวงจร และเขียนโปรแกรม																								
4	ทดลองโปรแกรม																								
5	จัดเตรียมวัสดุ อุปกรณ์																								
6	ทำชิ้นงานและติดตั้ง																								
7	ทดลองชิ้นงาน																								
8	สอบโครงงาน																								
9	นำเสนอ																								

3.3 รายการวัสดุอุปกรณ์ที่ใช้ในการทำโครงงานสิ่งประดิษฐ์

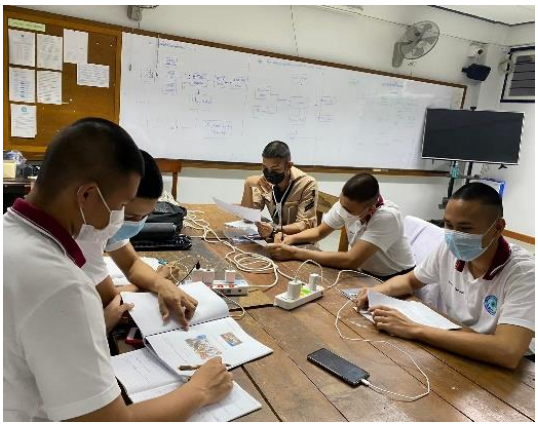
ลำดับ	รายการ	จำนวน	หน่วย	ราคา/หน่วย	รวม
1	เซ็นเซอร์วัดอุณหภูมิและความชื้น DHT11	1	ตัว	70	70
2	เซ็นเซอร์วัดความชื้นในดิน	1	ตัว	30	30
3	ปั๊มน้ำ 12V	1	อัน	130	130
4	เบรกเกอร์ 220Vac 20A	1	ตัว	120	120
5	เต้ารับไฟฟ้า 2 ช่อง	1	ตัว	130	130
6	หางปลาแฉกหุ้ม 2-3(WA)	4	ตัว	14	56
7	ตู้คอนโทรล ขนาด3.5 x 52 x 17(WA)	1	ตู้	400	400
8	สาย upload Arduino	1	เส้น	25	25
9	แผ่นอคิลิก ขนาด60 x 40cm.	3	แผ่น	85	255
10	รีเลย์ไฟ 5Vdc ควบคุมไฟ (2ช่อง)	1	ตัว	50	50
11	Switching Power supply 12Vdc 5A	1	ตัว	150	150
12	Adapter 9V	1	ตัว	50	50
13	AWG 22 ¼	1	เมตร	130	130
14	ท่อหดขนาด 4mm.	1	เมตร	8	8
15	ท่อหดขนาด 8mm.	1	เมตร	10	10
16	เทปพันสายไฟ ขนาด 1 x 1.5	1	ม้วน	10	10
17	สายไฟ V55 1.5	1	เมตร	50	50
18	สายยางใส	1	เมตร	12	12
19	ท่อกระดูกงู	1	เมตร	20	20
20	ท่อ PCV 1/4นิ้ว	3	เมตร	50	150
	รวม				**1,496**

งบประมาณที่ใช้ในการทำโครงงานสิ่งประดิษฐ์

เป็นเงินทั้งสิ้น 1,496 บาท (หนึ่งพันสี่ร้อยเก้าสิบหกบาทถ้วน)

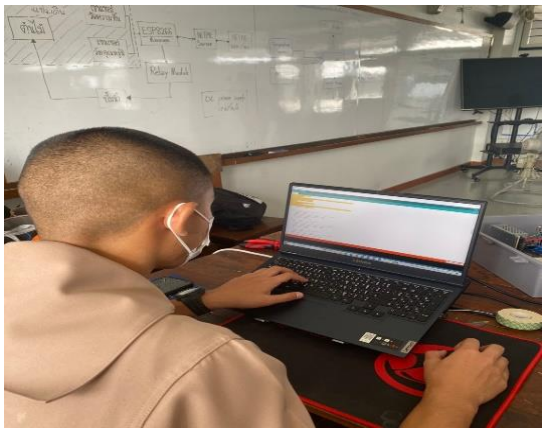
3.4 ขั้นตอนการดำเนินงาน

3.6.1 วางแผนออกแบบระบบรดน้ำต้นไม้และดูค่าสถานะผ่านเน็ตพาย



รูปที่ 3.4 วางแผนออกแบบ

3.6.2 เริ่มการเขียนโปรแกรมระบบรดน้ำต้นไม้และดูค่าสถานะผ่านเน็ตพาย

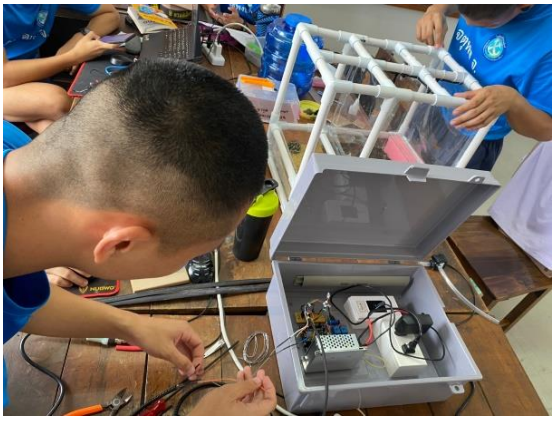


รูปที่ 3.5 การเขียนโปรแกรม

3.6.3 ทำการทดสอบวงจรระบบรดน้ำต้นไม้และดูค่าสถานะผ่านเน็ตพาย

รูปที่ 3.6 ทดสอบวงจร

3.6.4 นำแผงวงจรมาประกอบเขากับกลองใสอุปกรณ์



รูปที่ 3.7 นำแผงวงจรมาประกอบเขากับกลองใสอุปกรณ์

3.6.5 ทดลองนำกลองอุปกรณไปติดตั้งกับแปลงสาธิต

รูปที่ 3.8 นำกลองอุปกรณไปติดตั้งกับแปลงสาธิต

3.6.6 ทำการทดสอบการทำงานระบบรดน้ำต้นไม้และดูค่าสถานะผ่านเน็ตพาย

รูปที่ 3.9 ทดสอบการทำงานระบบรดน้ำต้นไม้และดูค่าสถานะผ่านเน็ตพาย

บทที่ 4
ผลการทดลอง

4.1 การทดลองระบบรดน้ำต้นไม้และดูค่าสถานะผ่านเน็ตพาย

การทดลอง					
ครั้ง	กำหนดค่าความชื้นในดิน		ความชื้นปัจจุบัน		ปั๊มน้ำ
	รดน้ำ	หยุดรดน้ำ	ก่อน	หลัง	
1	20%	70%	4%	78%	ทำงาน
2	30%	80%	31%	31%	ไม่ทำงาน
3	40%	90%	19%	95%	ทำงาน
4	50%	100%	42%	100%	ทำงาน
5	50%	70%	30%	92%	ทำงาน

สรุปผลการทดลอง

ระบบรดน้ำต้นไม้และดูค่าสถานะผ่านเน็ตพายจะทำงานเมื่อเซ็นเซอรวัดความชื้นในดิน (Soil Moisture Capacitive Sensor V2) ตรวจจับความชื้นในดินต่ำกว่าค่าที่กำหนดไว้ระบบจะทำการรดน้ำต้นไม้ และระบบจะหยุดการทำงานเมื่อความชื้นในดินมากกว่าค่าที่ที่กำหนดไว้ เช่น เซ็นเซอรวัดความชื้นในดิน (Soil Moisture Capacitive Sensor V2) ตรวจจับความชื้นในดินต่ำกว่า 20% ระบบจะทำการรดน้ำต้นไม้ และระบบจะหยุดการทำงานเมื่อความชื้นในดินมากกว่า 70% ระบบสามารถปรับค่าได้ตามต้องการและสามารถปรับระบบให้เป็น Manual แบบปิด/ปิดได้เมื่อระบบรดน้ำอัตโนมัติไม่ทำงาน

รูปที่ 4.2.1 แปลงสาธิตเครื่องรดน้ำต้นไม้

รูปที่4.2.2 หน้าจอ Display NETPIE

บทที่ 5
สรุป ปญหาและขอเสนอแนะ

5.1 สรุปโครงงาน

จากผลการดำเนินโครงงานสิ่งประดิษฐ์เรื่องระบบรดน้ำต้นไม้และดูค่าสถานะผ่านเน็ตพายถูกใช้ทดแทนการรดน้ำแบบเดิมได้ และยังลดการใช้ทรัพยากรโดยสิ้นเปลือง โดยระบบรดน้ำต้นไม้และดูค่าสถานะผ่านเน็ตพายสามารถทำงานตามฟังก์ชัน และโปรแกรมที่ถูกเขียนขึ้นและเหมาะที่จะนำไปใช้งานด้านอุตสาหกรรมรวมทั้งการเกษตรทั้งนี้โครงงานสำฤทธิ์ผลได้จากการทำงานเป็นหมู่คณะตลอดจนการร่วมกันคิดแก้ไขปัญหาข้อบกพร่องและสามารถนำโมเดลไปต่อยอดใช้ในการเกษตรได้จริงเพื่อลดค่าใช้จ่ายของเกษตรกร

5.2 ปัญหา

1.ในกรณีที่บอร์ด ESP8266 ไม่มีไฟเลี้ยงจะทำให้ไม่สามารถสั่งการรดน้ำได้

2.บอร์ด ESP8266 จำเป็นที่จะเชื่อมต่ออินเทอร์เน็ตตลอดเวลาเพื่อที่จะสั่งการรดน้ำ

3.Soil Moisture Capasitive Sensor ไม่เหมาะสมกับการนำมาวัดในพื้นที่ที่มีขนาดกว้าง

4.ถ้าต้องการรดน้ำในพื้นที่ ที่มีขนาดใหญ่จำเป็นจะต้องเปลี่ยนขนาดปั๊มน้ำและเพิ่ม Auxiliary Relay

5.3 ขอเสนอแนะ

ปญหาที่เกิดขึ้นในการจัดทำโครงงานชิ้นนี้ พบในสวนของเครื่องต่นแบบ ซึ่งทำให่เกิดข่อผิดพลาดในการทำงานของระบบการทำงาน ผู้จัดทำจึงขอเสนอแนวทางในการแกปญหา ดังตอไปนี้

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

2. โปรแกรม Arduino นั้นมีความซับซ้อน ผู้ศึกษาต้องใช้ความมุ่งมั่น และความพยายาม เพื่อทำความเข้าใจอย่างลึกซึ้งในการศึกษาโปรแกรม รวมไปถึงต้องทุ่มเทให้เวลาในการศึกษาโปรแกรม

3.ควรเพิ่มโมดูลตั้งเวลา เพื่อให้การรดน้ำมีประสิทธิภาพมากขึ้น สามารถกำหนดเวลาไม่ให้ปั๊มน้ำทำงานตอนอุณหภูมิสูง

4.ควรเปลี่ยนเป็นบอร์ด ESP32 ในกรณีที่ต้องการวัดความชื้นในพื้นที่ที่มีขนาดใหญ่

5.ควรเพิ่มฟังก์ชันในการแจ้งเตือนผ่านไลน์

บรรณานุกรม

โครงงานสิ่งประดิษฐ์นักเรียนจ่ารุ่น 61 เครื่องรดน้ำต้นไม้อัตโนมัติ

http://www.orade.com

Arduino Beafect HUB

หนังสือเชื่อมต่อและเขียนโปรแกรมสมองกล Arduino (ผู้เขียน รศ. ธีรวัฒน์ ประกอบผล)

www.researehgate.net

https://netpie.io

https://docs.netpie.io/getting-started.html

https://images.app.goo.gl/uwwDKef3ra422S5N9

https://images.app.goo.gl/y14ZuJJp8aj6JA1Y7

https://images.app.goo.gl/uu7VNKztqLX5F2717

https://images.app.goo.gl/LzJfmSnajJeuLuzE8

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

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

https://images.app.goo.gl/Btf9XNGHzd6BbZej8

https://images.app.goo.gl/E1tB3FbLT4L7fEXR9

https://images.app.goo.gl/9utGCjaVK8HeqUaY6

https://images.app.goo.gl/CPjKT6SCWQiDrYoP6

ภาคผนวก

โปรแกรมระบบแบบจำลองเครื่องรดน้ำ

โปรแกรมการทำงาน

```
#include <ESP8266WiFi.h>
#include <WiFiManager.h>
#include <PubSubClient.h>
#include <DHT.h>

const char* ssid = "Galaxy A52s 5GD90B";
const char* password = "szci4420";
const char* mqtt_server = "broker.netpie.io";
const int mqtt_port = 1883;
const char* mqtt_Client = "4424be42-fea8-4c15-a4b2-9814827474e0";
const char* mqtt_username = "mK2nnbVngZFGHXWp6BWnpuoQHsYVC7Kr";
const char* mqtt_password = "5wm(XX#1tATf3tf9r#agLMuGGA1g7IWo";
WiFiClient espClient;
PubSubClient client(espClient);
char msg[50];
int Modev;
int Relay1 = 0,Relay2 = 0;
int sensorPin = A0;
int ledPin = 16;
int value1;
int value2;
int Startch = 0;
String datastr;
#define relay1 16  // กำหนด D0 คือขา 16
#define R2 5  // กำหนด  D1 คือขา 5
#define D2 4
#define DHTPIN D2
#define DHTTYPE DHT11
DHT dht(DHTPIN,DHTTYPE);

void reconnect() {
  while (!client.connected()) {
    Serial.print("Attempting MQTT connection…");
    if (client.connect(mqtt_Client, mqtt_username, mqtt_password)) {
      Serial.println("connected");
      client.subscribe("@msg/#");
    } else {
      Serial.print("failed, rc=");
      Serial.print(client.state());
      Serial.println("try again in 5 seconds");
```

```
    delay(5000);
   }
  }
}
void callback(char* topic, byte* payload, unsigned int length) {
  Serial.print("Message arrived [");
  Serial.print(topic);
  Serial.print("] ");
  String message;
  String tpc;
  for (int i = 0; i < length; i++) {
   message = message + (char)payload[i];
  }
  Serial.println(message);
   if(String(topic) == "@msg/Relay1"){
     if (message == "on"){
      digitalWrite(relay1, 1);
      client.publish("@shadow/data/update", "{\"data\" : {\"Relay1\" : \"on\"}}");
      Serial.println("Relay ON");
      }
      else if (message == "off"){
      digitalWrite(relay1, 0);
      client.publish("@shadow/data/update", "{\"data\" : {\"Relay1\" : \"off\"}}");
      Serial.println("Relay OFF");
     }
  }
if (String(topic) == "@msg/Mode"){
    if (message == "1"){
      Modev = 1;
      client.publish("@shadow/data/update", "{\"data\" : {\"Mode\" : \"1\"}}");
      Serial.println("Aoto");
      Serial.println(Modev);
      }
      else if (message == "0"){
      Modev = 0;
      client.publish("@shadow/data/update", "{\"data\" : {\"Mode\" : \"0\"}}");
      Serial.println("Manual");
      Serial.println(Modev);
      }
    }
   else if (String(topic) == "@msg/value1"){
      value1 = message.toInt();
      datastr = "{\"data\" : {\"value1\" : " + String(value1)+ "}}";
      datastr.toCharArray(msg, (datastr.length() + 1));
      client.publish("@shadow/data/update", msg);
```

```
      Serial.println(value1);
      Serial.println(datastr);
      }
   else if (String(topic) == "@msg/value2"){
      value2 = message.toInt();
      datastr = "{\"data\" : {\"value2\" : " + String(value2)+ "}}";
      datastr.toCharArray(msg, (datastr.length() + 1));
      client.publish("@shadow/data/update", msg);
      Serial.println(value2);
      Serial.println(datastr);
      }else if (String(topic) == "@msg/reset1"){
     if (message == "reset1"){
      value1 = 0;
      value2 = 0;
      value1 = message.toInt();
      value2 = message.toInt();
      datastr = "{\"data\" : {\"value1\" : " + String(value1)+" ,\"value2\" : " + String(value2)+ "}}";
      datastr.toCharArray(msg, (datastr.length() + 1));
      client.publish("@shadow/data/update", msg);
      Serial.println(value1);
      Serial.println(value2);
      }
   }


}
void Start(){
  while (!client.connected()) {
  Serial.print("Sensor MQTT connection…");
  if (client.connect(mqtt_Client, mqtt_username, mqtt_password)) {
  Serial.println("NETPIE2020 connected");
  client.subscribe("@msg/#");
  client.publish("@shadow/data/update", "{\"data\" : {\"Mode\" : \"0\"}}");
  delay(2000);
  }
  }
}
  void onAutoConnWifi(String chk){
   WiFiManager wifiManager;
   if(chk=="reset"){
    wifiManager.resetSettings();
    Serial.println("reset wifi");
   }else{
    wifiManager.autoConnect("Galaxy A52s 5GD90B");
```

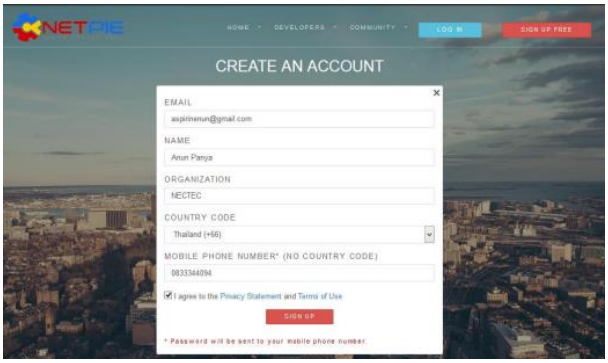
```
    Serial.println(WiFi.localIP());
   }
}
void setup() {
    pinMode (relay1, OUTPUT);
    digitalWrite(relay1,LOW);
    Serial.begin(115200);
    onAutoConnWifi("");
    WiFi.mode(WIFI_STA);
    while (WiFi.waitForConnectResult() != WL_CONNECTED){
      Serial.println("Connection Failed! Rebooting...");
      delay(5000);
      ESP.restart();
    }
    Serial.println("");
    Serial.println("Wi-Fi connected");
    Serial.print("IP Address : ");
    Serial.println(WiFi.localIP());
    client.setServer(mqtt_server, mqtt_port);
    client.setCallback(callback);
    dht.begin();

}

void loop() {
  if (!client.connected()) {
    reconnect();
  }
  if (Modev == 0){
    Malual();
    }else{
      auto1();
      }
  client.loop();
  delay(2000);
}
void Malual(){
  int Moisture;
  Moisture = analogRead(sensorPin);
  Moisture = map(Moisture, 727, 302, 0, 100);
  Serial.print("Soil moisture : ");
  Serial.print(Moisture);
  Serial.println(" %");
  if (!client.connected()) {
    reconnect();
  }
```

```
 client.loop();
 String data = "{\"data\": {\"Moisture\":" + String(Moisture) + "}}";
 Serial.println(data);
 data.toCharArray(msg, (data.length() + 1));
 client.publish("@shadow/data/update", msg);
 delay(2000);
  //------Check connection------
 if (!client.connected()) {
   reconnect();
 }
 client.loop();
 delay(2000);
 int humidity = dht.readHumidity();
 int temperature = dht.readTemperature();

  if (!client.connected()){
   reconnect();
 }
 client.loop();
 String data1 = "{\"data\": {\"humidity\":" + String(humidity) + ", \"temperature\":" +
String(temperature) + "}}";
 Serial.println(data1);
 data1.toCharArray(msg, (data1.length() + 1));
 client.publish("@shadow/data/update",msg);
 delay(2500);
 }

void auto1(){
 int Moisture;
 Moisture = analogRead(sensorPin);
 Moisture = map(Moisture, 727, 302, 0, 100);
 Serial.print("Soil moisture : ");
 Serial.print(Moisture);
 Serial.println(" %");
 String data = "{\"data\": {\"Moisture\":" + String(Moisture) + "}}";
 Serial.println(data);
 data.toCharArray(msg, (data.length() + 1));
 client.publish("@shadow/data/update", msg);
  int humidity = dht.readHumidity();
 int temperature = dht.readTemperature();
 client.loop();
 String data1 = "{\"data\": {\"humidity\":" + String(humidity) + ", \"temperature\":" +
String(temperature) + "}}";
 Serial.println(data1);
 data1.toCharArray(msg, (data1.length() + 1));
```

```
    client.publish("@shadow/data/update",msg);
    delay(2500);
  if (Moisture < value1){
    digitalWrite(relay1, 1);
    client.publish("@shadow/data/update", "{\"data\" : {\"Relay1\" : \"on\"}}");
    }else if (Moisture > value2){
    digitalWrite(relay1, 0);
    client.publish("@shadow/data/update", "{\"data\" : {\"Relay1\" : \"off\"}}");
    }
  }
```

การเข้าใช้งาน NETPIE

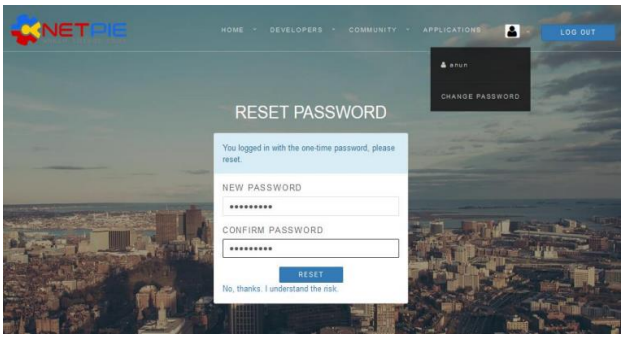
1. ไปที่เว็บไซต์ https//:netpie.io/sign up จะปรากฏหน้าเว็บดังรูป กรอกข้อมูลให้เรียบร้อย จากนั้นคลิกที่ปุ่ม SIGN UP เพื่อยืนยันการลงทะเบียน



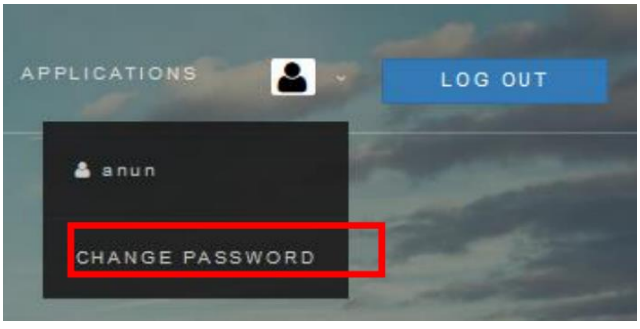
2. รอรับ SMS จากทาง NETPIE ซึ่งส่งไปยังหมายเลขโทรศัพท์เคลื่อนที่ที่ลงทะเบียนไว้
ตัวอย่าง SMS: **Your one-time password for NETPIE is 255906201550**

3. คลิกที่เมนู LOG IN เพื่อเข้าสู่ระบบนำอีเมลที่ลงทะเบียนไว้ใส่ในช่อง USERNAME OR EMAIL ADDRESS และนำรหัสผ่านที่ได้รับจาก SMS ใส่ในช่อง PASSWORD แล้วคลิกปุ่ม LOG IN

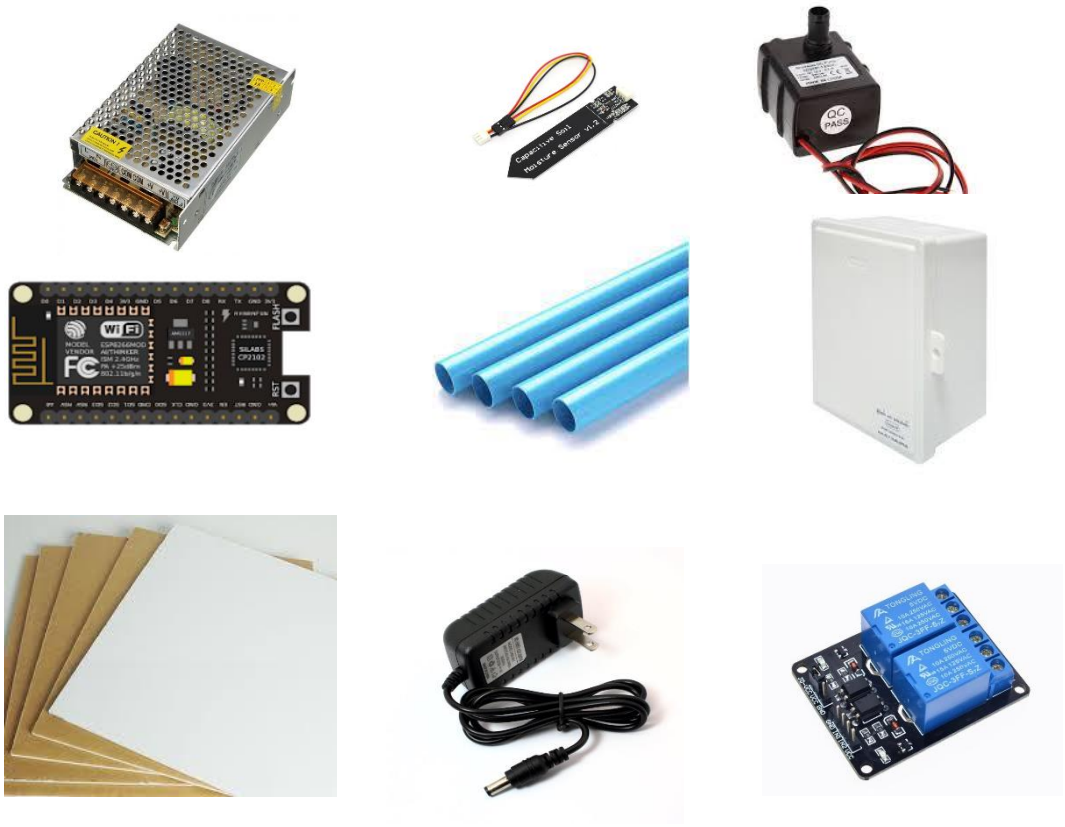
4. ตั้งรหัสผ่านใหม่โดยใส่รหัสผ่านใหม่ในช่อง NEW PASSWORD และ CONFIRM PASSWORD



5. สามารถเปลี่ยนรหัสผ่านได้ที่เมนูCHANGE PASSWORD



อุปกรณ์สำหรับการประกอบ



5.1 อุปกรณ์สำหรับการประกอบ

5.2 รูปแบบจำลองระบบรดน้ำต้นไม้และดูค่าสถานะผ่านเน็ตพาย

ประวัติคณะผู้จัดทำโครงงาน

นรจ.ภานุวัฒน์ จานุวี เหลา อิเล็กทรอนิกส์

ที่อยู่ 133 ซ.ลาดพร้าว 119 ถ.ลาดพร้าว แขวงคลองจั่น เขตบางกะปิ กรุงเทพ 10240

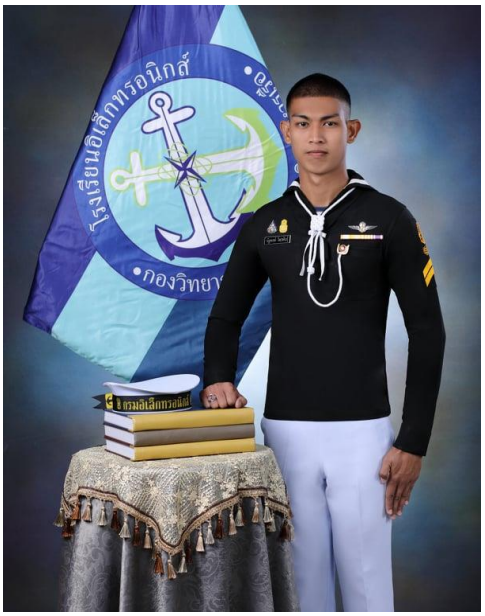
จบจาก โรงเรียนบางกะปิ

นรจ.ยศพัฒน์ คำแสน เหลา อิเล็กทรอนิกส์

ที่อยู 134/6 บ้านดอนไชย ต.ศิลาเพชร อ.ปัว จ.น่าน 55120

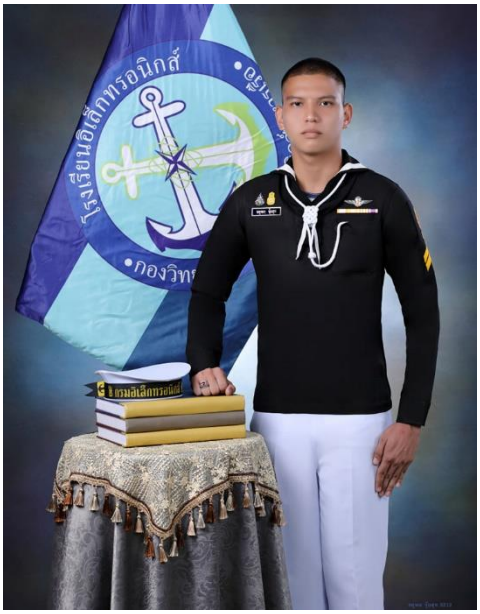
จบจาก โรงเรียนศรัทธาศิลาเพชรรังสรรค์

นรจ.นัฐพงษ์ ไตรพันธุ์ เหลา อิเล็กทรอนิกส์
ที่อยู่ 38/21ม.3 ต.บางครุ อ.พระประแดง จ.สมุทรปราการ 10130
จบจาก อิสลามวิทยาลัยแห่งประเทศไทย



นรจ.พันธกานต์ ไชยบุญทัน เหลา อิเล็กทรอนิกส์
ที่อยู่ 133 หมู่4 บ.โนนข่า ต.โนนทอง อ.แวงใหญ่ จ.ขอนแก่น 40330
จบจากโรงเรียนท่านางแนววิทยายน

นรจ.จตุพล จุ้ยสุข เหลา อิเล็กทรอนิกส์
ที่อยู่ 30/20 ถ.ราษฎ์พลี1ต.ชะอำ อ.ชะอำ จ.เพชรบุรี 76120
จบจาก โรงเรียนเบญจมเทพอุทิศจังหวัดเพชรบุรี



นรจ.จารุวิชย์ ทองชู เหลา อิเล็กทรอนิกส์
ที่อยู่ 18/3 ม.1 ต.วังวน อ.กันตัง จ.ตรัง 92110
จบจาก โรงเรียนจุ่งฮัวโซะเซียว จังหวัดตรัง