อินเตอร์เน็ตของทุกสรรพสิ่งของการควบคุมอุปกรณ์ไฟฟ้าและบันทึกข้อมูลการใช้พลังงาน

Internet of Things for Appliance control and energy data logger

นายธนวิชญ์ กมลฉ่ำ

นางสาวหทัยรัตน์ พินิจสุวรรณ

โครงงานวิศวกรรมนี้ เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมไฟฟ้า

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยศรีนครินทรวิโรฒ

ปีการศึกษา 2559

อินเตอร์เน็ตของทุกสรรพสิ่งของการควบคุมอุปกรณ์ไฟฟ้าและบันทึกข้อมูลการใช้พลังงาน

INTERNET OF THINGS FOR APPLIANCE CONTROL AND ENERGY DATA LOGGER

นายธนวิชญ์ กมลฉ่ำ

นางสาวหทัยรัตน์ พินิจสุวรรณ

โครงงานวิศวกรรมนี้ เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมไฟฟ้า

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยศรีนครินทรวิโรฒ

ปีการศึกษา 2559

ลิขสิทธิ์เป็นของคณะวิศวกรรมศาสตร์ มหาวิทยาลัยศรีนครินทรวิโรฒ

โครงงานวิศวกรรม

เรื่อง

อินเตอร์เน็ตของทุกสรรพสิ่งของการควบคุมอุปกรณ์ไฟฟ้าและบันทึกข้อมูลการใช้พลังงาน

ของ

นายธนวิชญ์ กมลฉ่ำ

นางสาวหทัยรัตน์ พินิจสุวรรณ

ได้รับอนุมัติจากคณะวิศวกรรมศาสตร์ให้นับเป็นส่วนหนึ่งของการศึกษาตามหลักสูตร

วิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมไฟฟ้า

ของมหาวิทยาลัยศรีนครินทรวิโรฒ

.....คณบดีคณะวิศวกรรมศาสตร์

(รองศาสตราจารย์ ดร.เวคิน ปิยรัตน์)

คณะกรรมการสอบโครงงานวิศวกรรม

.....ประธาน

(ผู้ช่วยศาสตราจารย์ ดร.ชาญไชย ไทยเจียม)

.....กรรมการ

(ผู้ช่วยศาสตราจารย์ ดร.นำคุณ ศรีสนิท)

.....กรรมการ

(ผู้ช่วยศาสตราจารย์ศิริพงษ์ ฉายสินธ์)

อินเตอร์เน็ตของทุกสรรพสิ่งของการควบคุมอุปกรณ์ไฟฟ้าและบันทึกข้อมูลการใช้พลังงาน
ปีการศึกษา 2559

โดย

นายธนวิชญ์ กมลฉ่ำ

นางสาวหทัยรัตน์ พินิจสุวรรณ

อาจารย์ที่ปรึกษา

ผู้ช่วยศาสตราจารย์ศิริพงษ์ ฉายสินธ์

บทคัดย่อ

จากสถิติการใช้ไฟฟ้าที่เพิ่มขึ้นเรื่อยๆ ในปัจจุบัน สาเหตุเกิดจาก อุปกรณ์เทคโนโลยีและการติดต่อสื่อสารต่างๆ มีการพัฒนาอย่างรวดเร็วและเข้ามามีบทบาทในการดำเนินชีวิตของมนุษย์เรามากกว่าแต่ก่อนมาก ส่งผลให้เกิดการใช้ทรัพยากรไฟฟ้าอย่างฟุ่มเฟือย เป็นปัญหาหนึ่งที่จำเป็นจะต้องมีการปรับปรุงแก้ไข และเมื่อกลุ่มผู้ใช้อินเตอร์เน็ตได้ขยายวงกว้างมากขึ้น อีกทั้งจากความแพร่หลายของบอร์ดคอมพิวเตอร์ขนาดจิ๋ว ตัวตรวจรู้ และมอดูลเชื่อมต่อการสื่อสารไร้สายต่างๆ รวมไปถึงการพัฒนาเทคโนโลยีอินเตอร์เน็ตของทุกสรรพสิ่ง เราจึงได้นำเทคโนโลยีอินเตอร์เน็ตของทุกสรรพสิ่ง มาใช้ให้เกิดประโยชน์ในการดำเนินชีวิตประจำวัน หรือเพื่อแก้ปัญหาต่างๆ ที่เกิดขึ้นในชีวิตประจำวัน ในโครงงานนี้จึงได้มีการนำเอาเทคโนโลยีอินเตอร์เน็ตของทุกสรรพสิ่ง มาประยุกต์ใช้กับ NodeMCU และตัวตรวจรู้เพื่อตรวจวัดพลังงานการใช้ไฟฟ้าและควบคุมการปิดเปิดอุปกรณ์ไฟฟ้า โดยในโครงงานนี้ได้แบ่งการทดลองออกเป็นสองส่วน ส่วนที่หนึ่ง คือ การควบคุมปิดเปิดอุปกรณ์ไฟฟ้าผ่านเว็บแอพพลิเคชั่น และส่วนที่สอง คือ การวัดพลังงานการใช้ไฟฟ้า โดยใช้ตัวตรวจรู้ PZEM-004T สำหรับวัดการใช้พลังงาน จากผลการทดลองพบว่าสามารถควบคุมปิดเปิดอุปกรณ์ไฟฟ้าและดูข้อมูลการใช้พลังงานไฟฟ้าผ่านเครือข่ายไร้สายได้

คำสำคัญ: อินเตอร์เน็ตของทุกสรรพสิ่ง NodeMCU ตัวตรวจรู้PZEM-004T

INTERNET OF THINGS FOR APPLIANCE CONTROL AND ENERGY DATA LOGGER

Academic Year 2016

By

Mister Thanavit Kamolchum

Miss Hathairat Pinitsuwan

Advisor

Asst.Prof.Siripong Chaysin

Abstract

The statistic of energy consumption has been increased because the technology and communication are rapidly developed. It caused the extravagant use of energy which need to be solved. Due to the increase of internet users, the prevalence of miniature computer boards, sensors and wireless communication interfaces modules, and the development of Internet of things, we have to apply the Internet of Things to use in daily life and to solve problems.

The objective of this project is to apply the Internet of Things (IoT) to NodeMCU, the sensors of the monitoring energy consumption and control appliance. The project has divided into two parts. The first part is appliance control through web application, and second part is monitoring energy consumption by PZEM-004T sensor for monitoring energy consumption. The results show that user can be control appliance and see data about energy consumption by using internet.

Keywords: Internet of Things (IoT), NodeMCU, PZEM-004T sensor

กิตติกรรมประกาศ

โครงงานวิศวกรรมศาสตร์นี้ สำเร็จลุล่วงได้ด้วยความช่วยเหลือจากผู้มีพระคุณหลายท่าน คณะผู้จัดทำโครงงานขอขอบคุณ ผู้ช่วยศาสตราจารย์ศิริพงษ์ ฉายสินธ์ อาจารย์ที่ปรึกษาโครงงาน ที่ได้กรุณาเสียสละเวลาอันมีค่าเพื่อให้คำปรึกษา คำแนะนำ ตรวจแก้ไขความเรียบร้อย ตลอดจนการชี้แนะในการหาคำตอบในปัญหาต่างๆ ระหว่างทำโครงงานนี้ ด้วยความเอาใจใส่อย่างยิ่ง

ขอขอบคุณผู้บริหารและพนักงานทุกคนของบริษัทตัวอย่าง ที่ทางคณะผู้จัดทำได้เข้าไปศึกษาวิจัยที่ท่านได้ให้ความร่วมมือในการให้ข้อมูล ความช่วยเหลือ และข้อเสนอแนะต่างๆ ให้กับคณะผู้จัดทำโครงงาน ทางคณะผู้จัดทำโครงงานขอขอบพระคุณเป็นอย่างสูง

ท้ายที่สุดนี้ทางคณะผู้จัดทำโครงงานขอขอบพระคุณทุกคนในครอบครัวที่ให้การสนับสนุนให้กำลังใจในการศึกษาตลอดมา และหวังเป็นอย่างยิ่งว่าโครงงานวิศวกรรมนี้ จะเป็นประโยชน์ต่อผู้ที่สนใจนำไปศึกษาไม่มากก็น้อยต่อไป ความดีและประโยชน์ใดๆ จากโครงงานวิศวกรรมนี้ ขอมอบให้กับผู้มีพระคุณทุกท่านที่ได้กล่าวมาทั้งหมด

คณะผู้จัดทำโครงงาน

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	ก
บทคัดย่อภาษาอังกฤษ	ข
กิตติกรรมประกาศ	ค
สารบัญ	ง
สารบัญตาราง	ฉ
สารบัญรูป	ช
บทที่ 1 บทนำ	1
1.1 ที่มาและความสำคัญของโครงงาน	1
1.2 วัตถุประสงค์ของโครงงาน	2
1.3 ขอบเขตของโครงงาน	2
1.4 ประโยชน์ที่คาดว่าจะได้รับ	2
บทที่ 2 ทฤษฎีที่เกี่ยวข้อง	3
2.1 ทฤษฎี	3
2.1.1 อินเตอร์เน็ตของทุกสรรพสิ่ง (Internet of Thongs: IoT)	3
2.1.2 รูปแบบการวัดกระแส	9
2.1.3 ตัวตรวจรู้ PZEM-004T	13
2.1.4 รีเลย์	17
2.1.5 NodeMCU	21
2.1.6 การประมวลผลแบบกลุ่มเมฆ (Cloud Computing)	23
2.1.7 เน็ตพาย (NETPIE)	28
2.1.8 แอนดรอยด์ (Android)	37
2.1.9 App Inventor	40
2.2 งานวิจัยที่เกี่ยวข้อง	41
บทที่ 3 วิธีการดำเนินงาน	43
3.1 ตัวควบคุมอุปกรณ์ไฟฟ้า	43
3.1.1 บล็อกไดอะแกรมตัวควบคุมอุปกรณ์ไฟฟ้า	43

สารบัญ(ต่อ)

	หน้า
3.1.2 ฮาร์ดแวร์	43
3.2 ตัวบันทึกข้อมูลการใช้พลังงาน	44
3.2.1 บล็อกไดอะแกรมตัวบันทึกข้อมูลการใช้พลังงาน	44
3.2.2 ฮาร์ดแวร์	44
3.3 ตัวควบคุมอุปกรณ์ไฟฟ้าและบันทึกข้อมูลการใช้พลังงาน	45
3.3.1 ภาพรวมของตัวควบคุมอุปกรณ์ไฟฟ้าและบันทึกข้อมูลการใช้พลังงาน	45
3.3.2 ฮาร์ดแวร์ในส่วนของภาคส่ง	46
3.3.3 เน็ตพาย (NETPIE)	47
บทที่ 4 ผลการดำเนินงาน	62
4.1 การแสดงผลในส่วนของฟรีบอร์ดของเน็ตพาย	62
4.1.1 ตัวควบคุมอุปกรณ์ไฟฟ้า	62
4.1.2 การแสดงผลการบันทึกข้อมูลการใช้พลังงาน	63
4.2 การแสดงผลในแอพพลิเคชั่นบนระบบปฏิบัติการแอนดรอยด์	65
4.3 ผลการดำเนินงานในส่วนของตัวควบคุมอุปกรณ์ไฟฟ้า	70
4.4 ผลการดำเนินงานในส่วนของการบันทึกข้อมูลการใช้พลังงาน	78
บทที่ 5 สรุปผล และข้อเสนอแนะ	82
5.1 สรุปผล	82
5.2 ข้อเสนอแนะ	83
เอกสารอ้างอิง	84
ประวัติย่อผู้ทำโครงงาน	85

สารบัญตาราง

ตารางที่ **หน้า**

2.1 โปรโตคอลการสื่อสารของมอดูล 14

2.2 ขาสัญญาณและการเชื่อมต่อ 21

2.3 อัตราค่าบริการ 30

2.4 อุปกรณ์ต่าง ๆ ที่ไมโครเกียร์ (Microgear) รองรับและเน็ตพาย (NETPIE) สามารถประมวลผลได้ 35

4.1 การควบคุมหลอดไฟดวงที่ 1 ผ่านเว็บฟรีบอร์ดของเน็ตพาย 70

4.2 การควบคุมหลอดไฟดวงที่ 2 ผ่านเว็บฟรีบอร์ดของเน็ตพาย 72

4.3 การควบคุมหลอดไฟดวงที่ 1 ผ่านแอพพลิเคชั่น 74

4.4 การควบคุมหลอดไฟดวงที่ 2 ผ่านแอพพลิเคชั่น 76

4.5 การบันทึกข้อมูลการใช้ไฟฟ้าแบบแยกส่วนด้วย PZEM-004T 78

4.6 การบันทึกข้อมูลการใช้ไฟฟ้าแบบแยกส่วนด้วย Multifunction Mini Ammeter 79

4.7 การบันทึกข้อมูลการใช้ไฟฟ้าโดยรวมด้วย PZEM-004T 80

4.8 การบันทึกข้อมูลการใช้ไฟฟ้าโดยรวมด้วย Multifunction Mini Ammeter 81

สารบัญรูป

รูปที่ **หน้า**

2.1 Kevin Ashton ผู้ที่ถูกยกย่องให้เป็นบิดาของอินเตอร์เน็ตของทุกสรรพสิ่ง (Internet of Things: IoT) 5

2.2 ตัวอย่างโทโพโลยีเครือข่ายอินเตอร์เน็ตของทุกสรรพสิ่ง 5

2.3 ตัวอย่างโทโพโลยีเครือข่ายอินเตอร์เน็ตของทุกสรรพสิ่ง 8

2.4 วงจรเบื้องต้นวัดกระแสโดยใช้ตัวต้านทานแบบขนาน (R-Shunt) 9

2.5 วงจรเบื้องต้นวัดกระแสโดยใช้ตัวตรวจรู้ปรากฏการณ์ฮอลล์ 10

2.6 ตัวตรวจรู้ปรากฏการณ์ฮอลล์ภายในตระกูล ACS 10

2.7 การวัดกระแสโดยใช้หม้อแปลงกระแส 11

2.8 สัญลักษณ์หม้อแปลงกระแส 11

2.9 หม้อแปลงกระแสแบบต่าง ๆ 12

2.10 ตัวตรวจรู้ PZEM-004T 13

2.11 แผนภาพการเดินสายไฟ (Wiring diagram) 16

2.12 รีเลย์ที่ใช้งาน 17

2.13 สัญลักษณ์ของรีเลย์ 18

2.14 สัญลักษณ์ของรีเลย์แทนโครงสร้างรีเลย์ 18

2.15 สภาวะการทำงานของรีเลย์ 18

2.16 บอร์ดรีเลย์ขนาด 2 ช่อง 19

2.17 ขาสัญญาณ 20

2.18 NodeMCU Devkit 1.0 (ESP-12E) Version 2 22

2.19 NodeMCU Devkit 1.0 (ESP-12E) Version 2 23

2.20 ส่วนหนึ่งของแอพพลิเคชั่นสำหรับผู้บริโภคที่ใช้ประโยชน์จากเทคโนโลยีการประมวลผลกลุ่มเมฆ 24

2.21 รายละเอียดของบริการต่าง ๆ ภายในเครือข่ายประมวลผลกลุ่มเมฆ 25

2.22 รูปแบบของการเลือกใช้บริการการประมวลผลแบบกลุ่มเมฆ 28

2.23 วิธีการสื่อสารของสิ่งต่าง ๆ ผ่านเน็ตพาย 29

2.24 Trusted System 33

2.25 Third-party System 34

2.26 Hybrid System 34

สารบัญรูป(ต่อ)

รูปที่		หน้า
	2.27 บอร์ดอาร์ดูโน่ (Arduino board)	35
	2.28 บอร์ดราสเบอร์รี่พาย (Rasberry Pi board)	35
	2.29 โทรศัพท์เคลื่อนที่	35
	2.30 สิ่งของ (things)	36
	2.31 ESP8266	36
	2.32 NodeMCU	36
	2.33 Spark Core	36
	2.34 สิ่งของ (Things)	37
	2.35 ARM Embed	37
	2.36 โครงสร้างต่าง ๆ ที่เป็นองค์ประกอบสำคัญในระบบปฏิบัติการแอนดรอยด์	39
	2.37 ลักษณะการใช้งาน MIT App Inventor	40
	3.1 บล็อกไดอะแกรมควบคุมอุปกรณ์ไฟฟ้า	43
	3.2 การเชื่อมต่ออุปกรณ์ในส่วนของการควบคุมอุปกรณ์ไฟฟ้า	43
	3.3 บล็อกไดอะแกรมตัวบันทึกข้อมูลการใช้พลังงาน	44
	3.4 การเชื่อมต่ออุปกรณ์ในส่วนของตัวบันทึกข้อมูลการใช้พลังงาน	44
	3.5 ภาพรวมของตัวควบคุมอุปกรณ์ไฟฟ้าและบันทึกข้อมูลการใช้พลังงาน	45
	3.6 การเชื่อมต่ออุปกรณ์ ฮาร์ดแวร์ในส่วนของภาคส่ง	46
	3.7 หน้าลงทะเบียนผู้ใช้งานเน็ตพาย (NETPIE)	47
	3.8 หน้าจัดการแอพพลิเคชั่นสำหรับผู้ใช้งานเน็ตพาย (NETPIE)	47
	3.9 ตัวอย่างหน้าแอพพลิเคชั่นหนึ่งของเน็ตพาย (NETPIE)	48
	3.10 การตั้งชื่อและเลือกชนิด Application Key	48
	3.11 รายละเอียดของคีย์	49
	3.12 เริ่มต้นการสร้างฟรีบอร์ดบนเน็ตพาย	50
	3.13 ตั้งชื่อ Freeboard ว่า ControlConsump	50
	3.14 เลือกประเภทของ Datasource เป็น NETPIE Microgear	50
	3.15 การระบุข้อมูลของ Datasource สำหรับใช้ในการควบคุม เปิด ปิด หลอดไฟดวงที่ 1	51
	3.16 การระบุข้อมูลของ Widget สำหรับสร้างปุ่มกดเปิดหลอดไฟดวงที่ 1	52

สารบัญรูป(ต่อ)

รูปที่		หน้า
3.17	การระบุข้อมูลของ Widget สำหรับสร้างปุ่มกดปิดหลอดไฟดวงที่ 1	53
3.18	การระบุข้อมูลของ Datasource สำหรับใช้ในการควบคุม เปิด ปิด หลอดไฟดวงที่ 2	54
3.19	การระบุข้อมูลของ Widget สำหรับสร้างปุ่มกดเปิดหลอดไฟดวงที่ 2	55
3.20	การระบุข้อมูลของ Widget สำหรับสร้างปุ่มกดปิดหลอดไฟดวงที่ 2	55
3.21	การระบุข้อมูลของ Datasource สำหรับใช้ในการบันทึกข้อมูลการใช้พลังงานไฟฟ้าแบบแยกส่วน	56
3.22	การระบุข้อมูลของ Widget สำหรับสร้างการแสดงผลค่าแรงดันไฟฟ้าที่วัดได้แบบแยกส่วน	57
3.23	การระบุข้อมูลของ Widget สำหรับสร้างการแสดงผลค่ากระแสไฟฟ้าที่วัดได้แบบแยกส่วน	58
3.24	การระบุข้อมูลของ Widget สำหรับสร้างการแสดงผลค่ากำลังไฟฟ้าที่วัดได้แบบแยกส่วน	58
3.25	การระบุข้อมูลของ Widget สำหรับสร้างการแสดงผลค่าพลังงานไฟฟ้าที่วัดได้แบบแยกส่วน	59
3.26	การระบุข้อมูลของ Datasource สำหรับใช้ในการบันทึกข้อมูลการใช้พลังงานไฟฟ้าแบบรวม	59
3.27	การระบุข้อมูลของ Widget สำหรับสร้างการแสดงผลค่าแรงดันไฟฟ้าที่วัดได้แบบรวม	60
3.28	การระบุข้อมูลของ Widget สำหรับสร้างการแสดงผลค่ากระแสไฟฟ้าที่วัดได้แบบแยกส่วน	60
3.29	การระบุข้อมูลของ Widget สำหรับสร้างการแสดงผลค่ากำลังไฟฟ้าที่วัดได้แบบรวม	61
3.30	การระบุข้อมูลของ Widget สำหรับสร้างการแสดงผลค่าพลังงานไฟฟ้าที่วัดได้แบบรวม	61
4.1	Widget สำหรับทำการควบคุมหลอดไฟ	62
4.2	Widget แสดงค่าการใช้พลังงานไฟฟ้าแบบแยกส่วน โดยแบ่งเป็น แรงดันไฟฟ้า กระแสไฟฟ้า กำลังงานไฟฟ้า และพลังงานไฟฟ้า	63

สารบัญรูป(ต่อ)

รูปที่		หน้า
	4.3 Widget แสดงค่าการใช้พลังงานไฟฟ้าแบบรวมโดยแบ่งเป็น แรงดันไฟฟ้า กระแสไฟฟ้า กำลังงานไฟฟ้า และพลังงานไฟฟ้า	64
	4.4 หน้าตาแอพพลิเคชั่น ชื่อ Lamphome	65
	4.5 หน้าจอหลักของแอพพลิเคชั่น Lamphome	66
	4.6 หน้าจอ LIGHT CONTROL	67
	4.7 หน้าจอแสดงสถานะ LIGHT CONTROL	67
	4.8 หน้าจอควบคุม ปิด เปิดหลอดไฟ โดยผ่านทางฟรีบอร์ดของเน็ตพาย ภายในแอพพลิเคชั่น	68
	4.9 หน้าจอสำหรับแสดงผลแรงดันไฟฟ้าและกระแสไฟฟ้าที่ใช้	69
	4.10 หน้าจอสำหรับแสดงผลกำลังงานไฟฟ้าและพลังงานไฟฟ้าที่ใช้	69

บทที่ 1
บทนำ

1.1 ที่มาและความสำคัญของโครงงาน

ในปัจจุบันถือเป็นยุคที่อินเตอร์เน็ต อุปกรณ์เทคโนโลยีและการติดต่อสื่อสารต่างๆ มีการพัฒนาอย่างรวดเร็วและเข้ามามีบทบาทต่อการดำเนินชีวิตของมนุษย์เรามากขึ้นกว่าเมื่อก่อนมาก การใช้เครือข่ายอินเตอร์เน็ตถือเป็นเครือข่ายที่มีประสิทธิภาพสูง ที่สามารถเชื่อมต่อคนทั้งโลกเข้าด้วยกันได้ และในอนาคตไม่เพียงแต่ผู้คนเท่านั้นที่จะถูกเชื่อมต่อเข้าด้วยกัน แต่อุปกรณ์เทคโนโลยีต่างๆ ก็จะถูกเชื่อมต่อเข้าด้วยกันเพื่อให้สามารถติดต่อสื่อสารกันได้เช่นกัน ซึ่งแนวคิดนี้ถูกเรียกว่า เทคโนโลยีอินเตอร์เน็ตของทุกสรรพสิ่ง (Internet of Things: IoT)

เมื่อกลุ่มผู้ใช้อินเตอร์เน็ตได้ขยายวงกว้างขึ้น อีกทั้งจากความแพร่หลายของอุปกรณ์สมองกลฝังตัว บอร์ดคอมพิวเตอร์ขนาดจิ๋ว ตัวตรวจรู้ (Sensor) และมอดูลเชื่อมต่อการสื่อสารไร้สายต่างๆ รวมไปถึงการพัฒนาเทคโนโลยีอินเตอร์เน็ตของทุกสรรพสิ่ง ไม่จำเป็นที่จะต้องเข้าไปร่วมวงในส่วนของผู้ผลิตต้นน้ำ เพราะในปัจจุบันการแข่งขันในการผลิตชิปมีค่อนข้างสูงและมีความหลากหลายให้เลือกใช้ จึงทำให้มีการผลิตออกมาเป็นบอร์ดสำเร็จรูปที่พร้อมสำหรับการพัฒนาพร้อมโค้ดตัวอย่าง ที่มุ่งหวังเพื่อให้นักพัฒนาสามารถนำไปพัฒนาต่อได้ง่าย ที่สำคัญบอร์ดเหล่านี้อยู่ในรูปแบบของฮาร์ดแวร์แบบเปิด มีผู้คนมากมายนำมาใช้พัฒนา และมีชุมชนเพื่อแลกเปลี่ยนข้อมูลความรู้ซึ่งกันและกัน และที่สำคัญอย่างมากอีกประการหนึ่งคือ ราคาของผลิตภัณฑ์เหล่านี้ เป็นราคาที่นักพัฒนาสามารถเข้าไปจับต้องได้ และสามารถที่จะซื้อหามาใช้งานได้ง่ายกว่าในอดีตเป็นอย่างมาก จึงทำให้เทคโนโลยีอินเตอร์เน็ตของทุกสรรพสิ่ง สามารถนำมาใช้งานได้จริงและเกิดประโยชน์ในการดำเนินชีวิตประจำวันหรือเพื่อแก้ไขปัญหาต่างๆ ในการทำงาน จากการที่สิ่งต่างๆ ถูกเชื่อมโยงเข้าสู่โลกอินเตอร์เน็ตทำให้มนุษย์สามารถสั่งการ ควบคุมการใช้งานอุปกรณ์ต่างๆ ผ่านทางเครือข่ายอินเตอร์เน็ต เราจึงเลือกใช้เทคโนโลยีอินเตอร์เน็ตของทุกสรรพสิ่ง มาประยุกต์ใช้กับอุปกรณ์สำหรับตรวจรู้ค่าพารามิเตอร์จากตัวตรวจรู้ เพื่อตรวจวัดพลังการใช้ไฟฟ้าภายในบ้านและควบคุมการเปิดปิดอุปกรณ์ไฟฟ้า เมื่อได้รับค่าพารามิเตอร์แล้วจะนำข้อมูลไปประมวลผลและส่งข้อมูลผ่านเครือข่ายอินเตอร์เน็ต เพื่อไปเก็บยังระบบคลาวด์ (Cloud) จากนั้นจะสามารถเรียกดูข้อมูลหรือควบคุมอุปกรณ์ไฟฟ้าผ่านเว็บแอพพลิเคชั่น ได้ในทุกที่ทุกเวลา ที่ผู้ใช้งานมีการเชื่อมต่ออินเตอร์เน็ต

ในโครงงานนี้ได้นำเอาการเชื่อมต่อแบบไร้สายมาใช้ โดยนำเอาอุปกรณ์เครื่องใช้ไฟฟ้ามาเชื่อมต่อกับไมโครคอนโทรลเลอร์และตัวตรวจรู้เพื่อรับข้อมูลการใช้พลังงานไฟฟ้า เราจะนำตัวตรวจรู้ไปตรวจวัดการใช้พลังงานในอุปกรณ์เครื่องใช้ไฟฟ้าที่เราต้องการและส่งข้อมูลไปเก็บไว้ยังคลาวด์ และเมื่อมีอุปกรณ์ที่เชื่อมต่อกับอินเตอร์เน็ตจะสามารถรับข้อมูลการใช้พลังงานและควบคุมการเปิดปิดอุปกรณ์ไฟฟ้าได้

1.2 วัตถุประสงค์ของโครงงาน

1.2.1 เพื่อประยุกต์การใช้งานตัวตรวจรู้ปริมาณการใช้ไฟฟ้าให้แสดงผลผ่านเว็บแอพพลิเคชั่น

1.2.2 เพื่อศึกษาและประยุกต์ใช้ไมโครคอนโทรลเลอร์ในการควบคุมอุปกรณ์ไฟฟ้า

1.2.3 เพื่อให้สามารถควบคุมอุปกรณ์ไฟฟ้าผ่านเว็บแอพพลิเคชั่นได้ทุกที่ทุกเวลา

1.2.4 เพื่อวัดพลังงานไฟฟ้าที่ใช้และแสดงผลผ่านเว็บแอพพลิเคชั่น

1.2.5 เพื่อเพิ่มความสะดวกและเป็นประโยชน์ต่อผู้ใช้ไฟฟ้า

1.3 ขอบเขตของโครงงาน

1.3.1 ตรวจวัดการใช้พลังงานไฟฟ้าด้วยตัวตรวจรู้

1.3.2 แสดงค่าปริมาณการใช้ไฟฟ้าผ่านเว็บแอพพลิเคชั่น

1.3.3 ใช้ไมโครคอนโทรลเลอร์ในการควบคุมเปิดปิดอุปกรณ์ไฟฟ้า

1.4 ประโยชน์ที่คาดว่าจะได้รับ

1.4.1 สามารถนำไมโครคอนโทรลเลอร์มาใช้ในการควบคุม เปิด ปิด อุปกรณ์ไฟฟ้า

1.4.2 สามารถพัฒนาความรู้ทางด้านไมโครคอนโทรลเลอร์ในการเชื่อมโยงกับเทคโนโลยีสมัยใหม่ โดยเชื่อมต่อผ่านทางระบบไร้สายและแสดงผลผ่านเว็บแอพพลิเคชั่น

1.4.3 ทำให้เกิดความสะดวกกับผู้ใช้อุปกรณ์ไฟฟ้า

บทที่ 2
ทฤษฎีที่เกี่ยวข้อง

2.1 ทฤษฎี

2.1.1 อินเตอร์เน็ตของทุกสรรพสิ่ง (Internet of Things: IoT)

ระบบเทคโนโลยี ที่อินเตอร์เน็ตของทุกสรรพสิ่งจะทำให้อุปกรณ์หรือสิ่งของต่างๆ สามารถเชื่อมต่อและสื่อสารผ่านระบบอินเตอร์เน็ตได้ โดยใช้แนวคิดการเชื่อมต่อของอุปกรณ์อัจฉริยะให้สื่อสารกันได้เอง โดยมนุษย์ไม่มีส่วนเกี่ยวข้องกับการทำงานแต่อย่างใด โดยมีเป้าหมายคือ ช่วยกันทำงานเพื่อให้คนสะดวกสบายมากยิ่งขึ้น

2.1.1.1 แนวคิดอินเตอร์เน็ตของทุกสรรพสิ่ง

Kevin Ashton ผู้ที่ถูกยกย่องให้เป็นบิดาของอินเตอร์เน็ตของทุกสรรพสิ่ง ในปี ค.ศ.1999 ซึ่งเขาเริ่มต้นโครงการ Auto-ID Center ที่สถาบันเทคโนโลยีแมสซาชูเซตส์ (Massachusetts Institute of Technology: MIT) จากเทคโนโลยี RFID ที่จะทำให้เป็นมาตรฐานระดับโลกสำหรับ RFID Sensors ต่างๆ ที่จะเชื่อมต่อกันได้ เมื่อสังเกตตัวอย่างการใช้งานข้างต้นของอินเตอร์เน็ตของทุกสรรพสิ่ง ซึ่งการใช้งานดังกล่าวจะใช้อินเตอร์เน็ตในการส่งหรือรับข้อมูลเป็นหลัก และในแต่ละกรณีแกตเจ็ตที่เชื่อมต่อกับอินเตอร์เน็ตไม่ได้เป็นเพียงคอมพิวเตอร์แท็บเล็ตหรือโทรศัพท์เท่านั้น แต่แกตเจ็ตเหล่านี้ถูกออกแบบใช้งานเพื่อวัตถุประสงค์ที่สำคัญ อันได้แก่ การสื่อสารระหว่างอุปกรณ์ต่างๆ เช่น เครื่องปรับอากาศ ตู้เย็น ไมโครเวฟ โทรศัพท์เคลื่อนที่ โทรศัพท์และอุปกรณ์อื่นๆ มีการเชื่อมต่อเข้าด้วยกัน โดยอุปกรณ์ที่ติดตั้งภายในบ้านดังกล่าวนี้สามารถควบคุมได้จากที่ทำงานหรือสถานที่ต่างๆ ผ่านเทคโนโลยีอินเตอร์เน็ต

ดังนั้น ไอเดียของอินเตอร์เน็ตของทุกสรรพสิ่ง จึงจำเป็นต้องใช้เทคโนโลยียูบีควิตัส (Ubiquitous Technology) หรือ ยูบิควิตัส คอมพิวติ้ง (Ubiquitous Computing) โดย Mark Welser นักวิทยาศาสตร์คอมพิวเตอร์แห่งศูนย์วิจัย Palo Alto ของบริษัท Xerox ประเทศสหรัฐอเมริกา ซึ่งภาษาละตินมีความหมายว่า มีอยู่ทุกหนทุกแห่ง โดยใช้แนวคิดความต้องการเชื่อมโยงระบบเครือข่ายการควบคุมอุปกรณ์และการสื่อสารข้อมูลกับอุปกรณ์และผลิตภัณฑ์เทคโนโลยี เพื่อให้สามารถเข้าถึงได้ทุกๆ สถานที่และทุกๆ เวลา ของกลุ่มผู้บริโภคหรือกลุ่มเป้าหมาย เพื่อตอบสนองความต้องการในการใช้งานผลิตภัณฑ์ที่มีเทคโนโลยีผสมผสานอยู่ โดยที่สามารถควบคุมผลิตภัณฑ์ที่มีอยู่ได้ตามความต้องการยูบิควิตัส คอมพิวติ้ง ตามแนวคิดของ Welser จะมีลักษณะเด่น 3 ประการ ได้แก่

(1) การเชื่อมต่อกับเครือข่าย (Network) หมายถึง การติดต่อสื่อสารได้ในทุกๆ ที่ ได้อย่างเหมาะสม สำหรับการใช้งานไปยังสถานที่ต่างๆ ซึ่งต้องใช้งานผ่านเครือข่ายคอมพิวเตอร์

(2) ผู้ใช้ต้องไม่มีความรู้สึกหรือรับรู้ว่าอุปกรณ์ดังกล่าวที่กำลังใช้งานอยู่เป็นส่วนใดส่วนหนึ่งของการใช้คอมพิวเตอร์อยู่ โดยเปรียบเสมือนที่ผู้ใช้กำลังใช้งานอุปกรณ์ต่างๆ ในชีวิตประจำวันเท่านั้น

(3) การให้บริการจะเปลี่ยนแปลงตามบริบท (Context) หมายถึง คอมพิวเตอร์จะให้บริการแต่ละบุคคลแตกต่างกันไปตามสถานการณ์ โดยขึ้นอยู่กับตำแหน่งที่อยู่ของผู้ใช้ คุณลักษณะเฉพาะตัว (ID) ของผู้ใช้งานและอุปกรณ์ (Device) การใช้งานต่างๆ ที่เป็นปัจจัยทางกายภาพ เช่น เวลา อุณหภูมิ เป็นต้น

นอกจากนี้ การหาขีดจำกัดของความก้าวหน้าทางเทคโนโลยีอย่างไม่หยุดยั้ง ตามคำอธิบายสิ่งที่เกี่ยวข้องกับเทคโนโลยีของนักเขียนนิยายวิทยาศาสตร์ ชื่อ อาร์เธอร์ซีคลาร์ก (Arthur C.Clarke) ได้อธิบายกฎข้อที่ 2 ที่ใช้อธิบายเทคโนโลยี วิธีเดียวที่จะทราบขีดจำกัดของความเป็นไปได้ก็คือพยายามก้าวไปให้ไกลเกินขีดจำกัดนั้นไปสู่ความเป็นไปได้ (The Only Way of Discovering the Limits of the Possible Is to Venture a Little Way Past Them into the Impossible) และกฎข้อที่ 3 ที่กล่าวไว้ว่า เทคโนโลยีใดๆ ที่ล้ำยุคเพียงพอก็จะไม่แตกต่างไปจากความประหลาดมหัศจรรย์ ที่เนรมิตขึ้นมาด้วยเวทมนต์คาถา (Any Sufficiently Advanced Technology is Indistinguishable from Magic) จะมีการทำงานที่อัจฉริยะมากยิ่งขึ้น (Smart Products) ที่สามารถแบ่งออกเป็น 3 ส่วนประกอบหลักๆ ได้แก่ โครงสร้างของสินค้า (Physical Component) ส่วนประกอบอัจฉริยะ (Smart Component) และส่วนของการเชื่อมต่อ (Connectivity Component) โดย Harvard Business Review ได้อธิบายรายละเอียดของส่วนต่างๆ ไว้ดังนี้

(1) โครงสร้างของสินค้า (Physical Component)

คือ ส่วนของแผงวงจรอิเล็กทรอนิกส์และไฟฟ้า แหล่งจ่ายไฟฟ้า เป็นต้น

(2) ส่วนประกอบอัจฉริยะ (Smart Component)

ได้แก่ ส่วนที่เป็นอุปกรณ์ตรวจจับจำพวกตัวตรวจรู้ต่างๆ อุปกรณ์เก็บข้อมูล และส่วนของโปรแกรม ได้แก่ ระบบปฏิบัติการ (OS) เป็นต้น

(3) ส่วนของการเชื่อมต่อ (Connectivity)

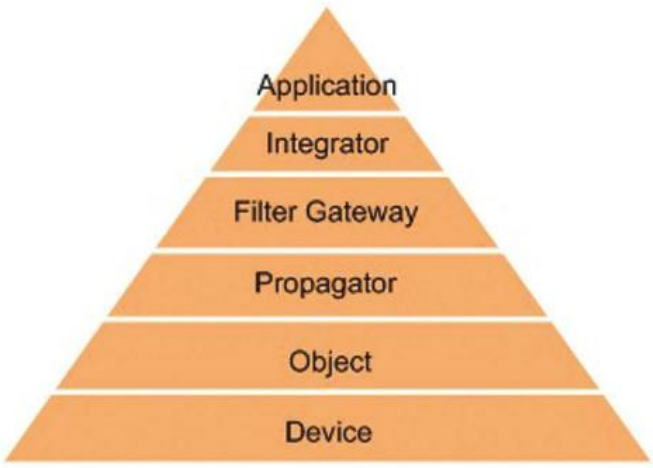
คือ ส่วนที่ใช้สื่อสารเชื่อมต่อและการแลกเปลี่ยนข้อมูลของอุปกรณ์ ซึ่งจะสามารถเป็นแบบ One-to-One, One-to-Many หรือ Many-to-Many เป็นต้น

รูปที่ 2.1 Kevin Ashton ผู้ที่ถูกยกย่องให้เป็นบิดาของอินเตอร์เน็ตของทุกสรรพสิ่ง (Internet of Things: IoT)

ที่มา: http://www.nrgm.nl/in-print/kevin-ashton-iot-revisited/

2.1.1.2 องค์ประกอบลอจิคัลในลำดับการเชื่อมโยงของระบบอินเตอร์เน็ตของทุกสรรพสิ่ง

ในระหว่างที่มาตรฐานอุปกรณ์ของระบบอินเตอร์เน็ตของทุกสรรพสิ่ง กำลังได้รับการพัฒนาจากมุมมองฟังก์ชันขององค์ประกอบด้วยลำดับชั้นลอจิคัลทั้งหมด 7 ชั้น ซึ่งองค์ประกอบในแต่ละชั้นอาจเป็นซอฟต์แวร์หรือฮาร์ดแวร์ก็ได้



รูปที่ 2.2 ตัวอย่างโทโพโลยีเครือข่ายอินเตอร์เน็ตของทุกสรรพสิ่ง

ที่มา: พงศ์พันธุ์ ปริยวงศ์. เซมิคอนดักเตอร์ อิเล็กทรอนิกส์. 2559; 419: 80.

(1) ชั้นอุปกรณ์ (Device) ชั้นนี้เป็นชั้นอุปกรณ์จำพวกตัวตรวจรู้ หรือตัวขับเร้า (Actuator) จะทำหน้าที่แปลงค่าจากปริมาณกายภาพให้เป็นสัญญาณไฟฟ้า เช่น ตัวตรวจรู้ที่วัดการเคลื่อนที่ ความดัน ความชื้น อัตราการเต้นหัวใจ ระดับก๊าซ โดยการใช้งานส่วนใหญ่จะนำตัวตรวจรู้ไปติดตั้งกับวัตถุ (Object) เพื่อให้มีการทำงานร่วมกับส่วนอื่น ในบางกรณีจะมีเงื่อนไขของหน่วยความจำเข้ามาเกี่ยวข้องสำหรับบันทึกข้อมูลและแจ้งเตือน เช่น ตัวตรวจรู้ที่ติดตั้งในยานพาหนะเพื่อแจ้งข้อมูลความเร็ว และตำแหน่งให้กับยานพาหนะที่อยู่ใกล้เคียง

ส่วนตัวขับเร้าจะรับคำสั่งที่ผ่านมาทางเกตเวย์ (Gateway) แล้วดำเนินการตามคำสั่งด้วยการเปลี่ยนสถานะภายในหรือสถานะของการทำงาน ตัวอย่างเช่น มีคำสั่งที่มีวัตถุประสงค์เพื่อรักษาระยะห่างกับรถคันหน้าส่งมายังตัวขับเร้า ซึ่งตัวขับเร้าก็จะทำงานโดยตัดแหล่งจ่ายเชื้อเพลิงที่ส่งไปยังเครื่องยนต์ ทำให้รถช้าลง ระยะห่างกับรถคันหน้าก็จะเพิ่มขึ้นจนถึงระดับปลอดภัยตามคำสั่งที่ได้รับมา หรือกรณีของการควบคุมมอเตอร์ด้วยแสง ตัวขับเร้าอาจเป็นไดโอดที่ไวต่อแสงหรือเป็นชิฟควบคุม

(2) ชั้นวัตถุ (Object) คือแพ็กเกจของอุปกรณ์ซึ่งเป็นแหล่งกำเนิดของข้อมูล โดยสามารถแบ่งวัตถุออกเป็น 2 โหมด คือ โหมดพาสซีฟ (Passive) และโหมดแอกทีฟ (Active) ในกรณีของโหมดพาสซีฟ ภายในวัตถุจะบรรจุหน่วยชี้วัด เช่น บาร์โค้ด (Barcode) ซิปเซ็ต RFID หรือชิ้นส่วนอื่นที่สามารถอ่านได้หรือระบุได้ ในโหมดนี้ยังสามารถเพิ่มเนื้อหาของการติดตั้งวัตถุ ที่ตำแหน่งนั้นหรือที่เวลานั้น ส่วนโหมดแอกทีฟ เป็นโหมดของวัตถุซึ่งภายในมีการติดตั้งชิ้นส่วนอิเล็กทรอนิกส์เพื่อดำเนินการ โดยชิ้นส่วนนี้อาจเป็น ตัวตรวจรู้ เกตเวย์ หรือตัวขับเร้า

ในกรณีของการนำวัตถุ ไปใช้ในกิจกรรมอื่นนอกเหนือจากฟังก์ชันปกติ ก็จำเป็นต้องเพิ่มชุดองค์ประกอบแกนหลัก (Core Component) อื่นเข้าไป เพื่อให้เป็นไปตามเป้าหมายทางธุรกิจหรือเงื่อนไขทางเทคนิค ตัวอย่างของวัตถุที่นำมาใช้งานกับระบบอินเตอร์เน็ตของทุกสรรพสิ่ง เช่น ชุดรักษาความปลอดภัยสำหรับบ้าน ภายในจะประกอบด้วย ตัวตรวจรู้อุณหภูมิ ตัวตรวจรู้ตรวจจับควัน และอื่นๆ หรือชุดควบคุมปริมาณเชื้อเพลิงในรถยนต์ ที่มีการเดินสายเตรียมไว้เพื่อส่งสัญญาณไปยังตัวขับเร้า เพื่อปรับลดปริมาณเชื้อเพลิงที่ส่งไปยังเครื่องยนต์

(3) ชั้นการแพร่กระจาย (Propagator) เป็นชั้นที่มีจุดประสงค์เพื่อติดต่อกับเครือข่าย เป็นตัวกลางการเชื่อมต่อระหว่างวัตถุกับเกตเวย์ ทำให้วัตถุสามารถเข้าถึงเครือข่ายได้ผ่านทางซอฟต์แวร์ โดยการแพร่กระจาย จะบริหารการรวบรวมข้อมูลและกระจายข้อความไปยังวัตถุต่างๆ เพื่อการส่งข้อมูลทางไกลการตอบรับข้อมูลและการร้องขอข้อมูล นอกจากนี้ จากการที่เครือข่ายอาจมีการใช้โพรโคตอลมากกว่า 1 โพรโทคอล เช่นโพรโทคอลที่ใช้กับการเชื่อมโยงทำให้การแพร่กระจาย ต้องรองรับการแปลงทราฟฟิกข้อมูลระหว่างโพรโทคอลที่แตกต่างกันด้วย เช่น โพรโทคอลไร้สายแบบระยะใกล้กับระยะไกลสำหรับการเชื่อมโยงโหนด การแพร่กระจายอาจมีการต่อไปยังอินเตอร์เน็ตหรือไม่ก็ได้ ขึ้นอยู่กับ

โพรโทคอลเครือข่ายที่ใช้ ตำแหน่งหรือลำดับที่อยู่ในการเชื่อมโยง รวมถึงรูปแบบของการใช้งานของวัตถุว่าเป็นแบบอยู่กับที่หรือเคลื่อนที่

ในกรณีที่โหนดการแพร่กระจาย ซึ่งอาจติดตั้งในซอฟต์แวร์ เชื่อมต่ออยู่กับอินเตอร์เน็ตเพื่อต่อไปยังฟิลเตอร์เกตเวย์ (Filter gateway) โหนดการแพร่กระจายนี้อาจเป็นซอฟต์แวร์การใช้งานที่ทำงานบนฮาร์ดแวร์มาตรฐาน ซึ่งใช้ระบบปฏิบัติการเพื่อรับบริการจากเว็บ สำหรับใช้กับไคลเอนท์ (Client) และเซิฟเวอร์ที่อยู่ไกลซึ่งทำหน้าที่เป็นเกตเวย์ และตัวอินทิเกรต (Integrator) โดยโหนดการแพร่กระจายสามารถสร้างการเชื่อมต่อด้านหนึ่งไปยังโครงข่ายสื่อสารพื้นฐาน และอีกด้านเป็นการเชื่อมต่อกับคลาวด์ (Cloud) ของอีกระบบ

โหนดการแพร่กระจาย อาจเป็นโหนดเครือข่ายท้องถิ่นที่ถูกใช้เพื่อสนับสนุนวัตถุต่างๆ โดยเฉพาะ เช่น Security hub ซึ่งซอฟต์แวร์ของไคลเอนท์ สามารถนำมาติดตั้งบนอุปกรณ์สมาร์ท (Smart device) พีซี แทบเลต หรือโทรศัพท์เคลื่อนที่ก็ได้ ถ้าฮาร์ดแวร์อุปกรณ์เหล่านี้รองรับ ตัวอย่างการใช้งานโหนดการแพร่กระจาย เช่น การใช้เป็นเซิฟเวอร์คอยติดตามปริมาณกายภาพ ได้แก่ อุณหภูมิ ปริมาณฝน ความชื้นในดิน ภายในพื้นที่ฟาร์มหรือบริเวณใด บริเวณหนึ่ง โดยโหนดจะรวบรวมข้อมูล รวมถึงควบคุมติดตามอุปกรณ์ในพื้นที่รับผิดชอบ

(4) ชั้นฟิลเตอร์เกตเวย์ (Filter gateway) เป็นชั้นที่คอยตรวจสอบการส่งข้อมูลไปยังตัวอินทิเกรตในการออกแบบ เกตเวย์โดยทั่วไปจะเชื่อมต่ออยู่กับอินเตอร์เน็ตและสามารถตั้งอยู่ที่ไหนก็ได้ โดยมีฟิลเตอร์ ทำหน้าที่ดึงข้อมูลสำคัญหรือข้อมูลที่ต้องการจากข้อมูลทั้งหมดที่ถูกรวบรวมและส่งมาทางการแพร่กระจาย ซึ่งควบคุมอุปกรณ์ ในการตั้งค่าฟิลเตอร์สามารถกำหนดให้สร้างเส้นทางการส่งข้อมูลประเภทอื่นไปยังตัวอินทิเกรตที่แตกต่างกันได้ นอกจากนี้ฟิลเตอร์ยังสามารถทำหน้าที่เป็นไฟร์วอล (Firewall) เพื่อป้องกันระบบและการใช้งานจากการโจมตีภายนอก

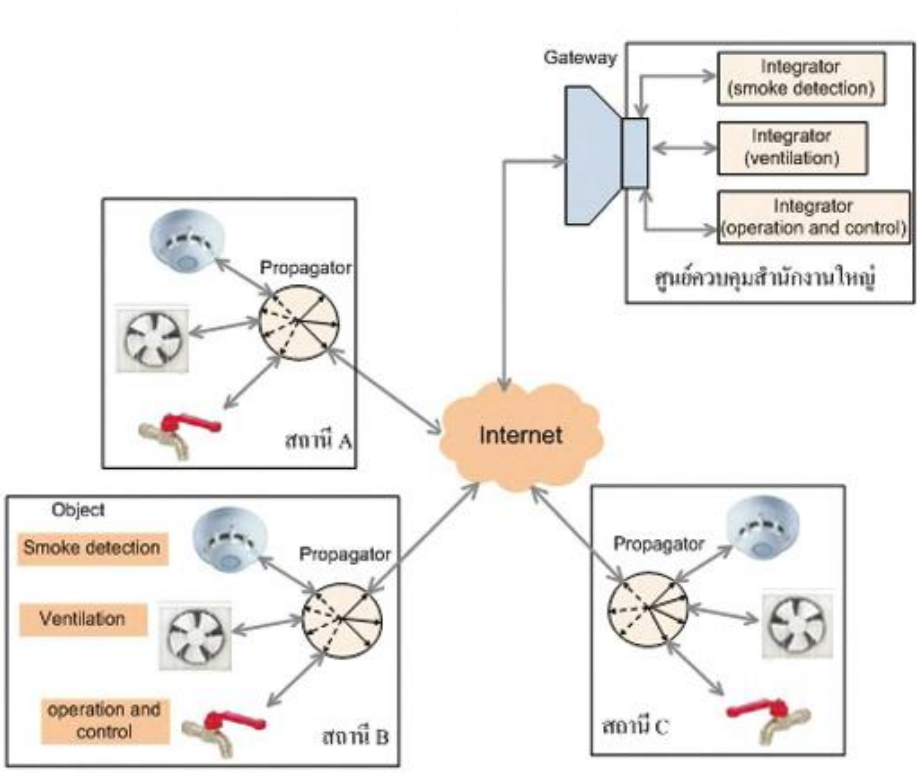
(5) ชั้นตัวอินทิเกรต (Integrator) เมื่อพิจารณาถึงหน้าที่ของการแพร่กระจาย ซึ่งรวบรวมข้อมูลตามเงื่อนไขที่กำหนด ในด้านของตัวอินทิเกรตก็จะทำหน้าที่รวมข้อมูลจากหลายแหล่งเข้าด้วยกันบนเงื่อนไขที่กว้างขึ้น เพื่อพิจารณาเป็นภาพรวมของการนำไปใช้งาน เช่น การรวมข้อมูลสภาพอากาศในระยะยาวที่เก็บได้จากฟาร์มกับตารางเวลาการเพาะปลูก สำหรับการต่อยอดจากข้อมูลเหล่านี้ ก็จะเน้นเรื่องของการวิเคราะห์และประมวลผลนอกจากนี้ตัวอินทิเกรตยังมีหน้าที่เตรียมพารามิเตอร์การตั้งค่าให้กับฟิลเตอร์เกตเวย์ เพื่อใช้ในการเลือกข้อมูลที่ถูกต้องมาตามเส้นทาง

(6) ชั้นแอพพลิเคชั่น (Application) เป็นชั้นการใช้งานของอินเตอร์เน็ตของทุกสรรพสิ่งจากข้อมูลทั้งหมดที่ได้รับมา การแสดงผลจะผ่านส่วนที่แรกว่าพรีเซนเตอร์ (Presenter) ซึ่งจะนำเสนอข้อมูลต่อผู้ที่เกี่ยวข้องในรูปของแสง เสียง หรือผ่านจอแสดงผล เช่น ระบบควบคุมทราฟฟิกกลางที่ดำเนินการโดยผู้ดูแลเมือง แอพพลิเคชั่นที่ติดตั้งในรถยนต์จะแจ้งข้อมูลผ่านหน้าจอเพื่อให้คนขับใช้ในการตัดสินใจ แอพพลิเคชั่นอาจเน้นเป็นระบบขนาดใหญ่ที่ทำงานอยู่ในศูนย์ข้อมูล เพื่อลดค่าใช้จ่ายและการใช้

ทรัพยากรของแอพพลิเคชั่นที่ไปติดตั้งในเทอร์มินอล ซึ่งสามารถเข้าถึงศูนย์ข้อมูลผ่านบริการเว็บ การนำแบบจำลององค์ประกอบนี้มาใช้ จะช่วยให้เกิดความเข้าใจสถาปัตยกรรมของระบบมากขึ้น

สำหรับการใช้งานหลายกรณีจะมีการใส่อุปกรณ์หลายชิ้นลงในวัตถุต่างๆ ซึ่งแต่ละอุปกรณ์จะมีการแบ่งปันฟังก์ชันการแพร่กระจาย เพื่อเชื่อมต่อกับเครือข่าย หรือกรณีของระบบรักษาความปลอดภัยภายในบ้าน ชุดของตัวตรวจรู้จะถูกแพ็กอยู่ในวัตถุ แล้วกระจายติดตั้งไปตามตำแหน่งต่างๆ ภายในบ้าน ส่วนตัวฮับ (Hub) ก็จะประกอบด้วย การแพร่กระจาย ฟิลเตอร์เกตเวย์ ตัวอินทิเกรต และแอพพลิเคชั่น

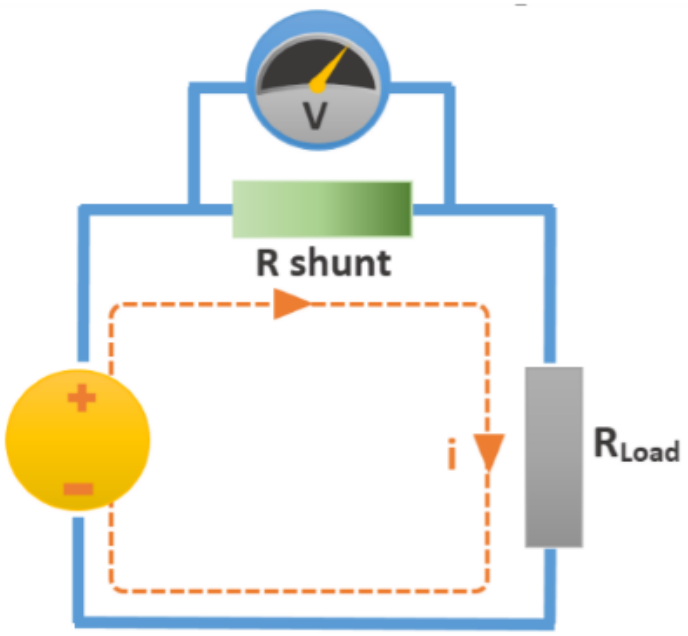
ตัวอย่างโทโพโลยีทางกายภาพของเครือข่ายอินเตอร์เน็ตของทุกสรรพสิ่ง ซึ่งแบ่งเป็นการรับสัญญาณจาก 3 วัตถุ ได้แก่ ตัวตรวจจับควัน (Smoke detector) พัดลมไหลเวียนอากาศ (Ventilation) และชุดดำเนินการควบคุม (Operation and control) เช่น วาล์วน้ำ ซึ่งทั้งหมดจะติดตั้งอยู่ในสถานี A, B และ C โดยในแต่ละสถานีจะมีการเชื่อมโยงวัตถุไปหาโหนดการแพร่กระจาย ที่อาจเชื่อมต่อผ่านสายหรือผ่านเครือข่ายไร้สายก็ได้ จากนั้นโหนดการแพร่กระจายก็จะเชื่อมโยงผ่านเครือข่ายอินเตอร์เน็ตไปยังเกตเวย์ที่ศูนย์ควบคุมสำนักงานใหญ่ ที่มีการติดตั้งตัวอินทิเกรตของแต่ละส่วน เพื่อรวมข้อมูลหลายแหล่งเข้าด้วยกันบนเงื่อนไขการดำเนินการที่กว้างขึ้น



รูปที่ 2.3 ตัวอย่างโทโพโลยีเครือข่ายอินเตอร์เน็ตของทุกสรรพสิ่ง
ที่มา: พงศ์พันธุ์ ปริยวงศ์. เซมิคอนดักเตอร์ อิเล็กทรอนิกส์. 2559; 419: 81.

2.1.2 รูปแบบการวัดกระแส

2.1.2.1 การตรวจวัดกระแสด้วยตัวต้านทาน (Current Sensing Resistor)

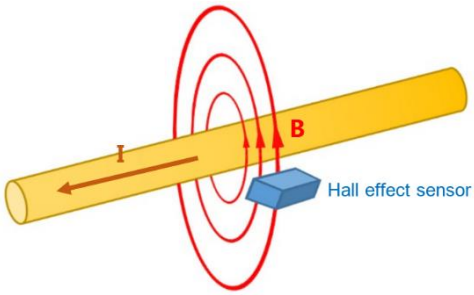


รูปที่ 2.4 วงจรเบื้องต้นวัดกระแสโดยใช้ตัวต้านทานแบบขนาน (R-Shunt)
ที่มา: http://www.thaieasyelec.com/article-wiki/review-product-article/

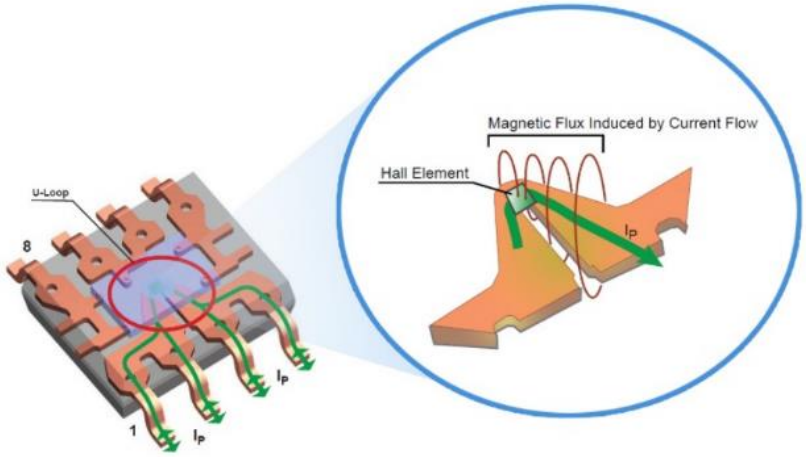
เป็นการวัดกระแสโดยใช้ตัวต้านทานแบบขนาน (R-Shunt) โดยอาศัยหลักการ คือ วัดแรงดันที่ตกคร่อมตัวต้านทานค่าน้อยๆ ซึ่งต่ออนุกรมกับตัวต้านทานโหลด (R_{Load}) เรียกว่าตัวต้านทานแบบขนาน (R_{Shunt}) โดยสามารถวัดได้ทั้งไฟดีซี หรือไฟกระแสตรง และเอซี หรือไฟกระแสสลับ และคำนวณกลับเป็นกระแสจากสูตร

$$I = \frac{V_{shunt}}{R_{shunt}} \tag{2.1}$$

2.1.2.2 ตัวตรวจรู้ปรากฏการณ์ฮอลล์ (Hall Effect Sensor)



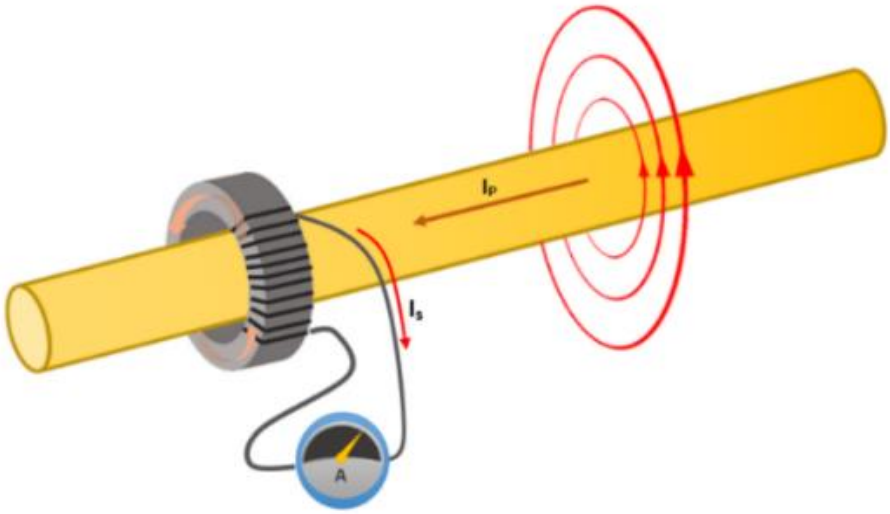
รูปที่ 2.5 วงจรเบื้องต้นวัดกระแสโดยใช้ตัวตรวจรู้ปรากฏการณ์ฮอลล์
ที่มา: http://www.thaieasyelec.com/article-wiki/review-product-article/



รูปที่ 2.6 ตัวตรวจรู้ปรากฏการณ์ฮอลล์ภายในตระกูล ACS
ที่มา: http://www.thaieasyelec.com/article-wiki/review-product-article/

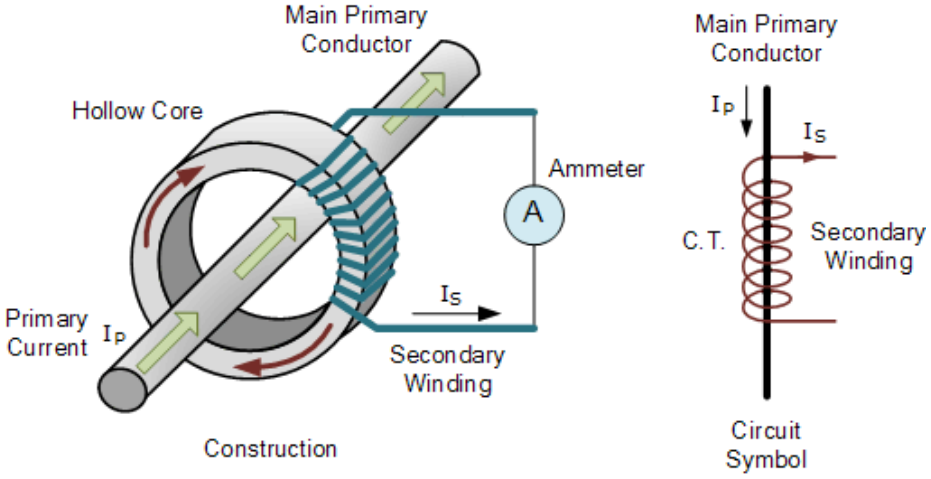
ตัวตรวจรู้ปรากฏการณ์ฮอลล์เป็นการวัดกระแสทางอ้อม เมื่อเราจ่ายกระแสไฟฟ้าทั้งกระแสตรง และกระแสสลับ จะทำให้เกิดเส้นสนามแม่เหล็กรอบสายไฟ เมื่อตัวตรวจรู้ปรากฏการณ์ฮอลล์อยู่ในบริเวณเส้นสนามแม่เหล็กของสายไฟ จะส่งสัญญาณออกมาตามระดับสนามแม่เหล็กที่วัดได้

2.1.2.3 หม้อแปลงกระแส (Current Transformer)



รูปที่ 2.7 การวัดกระแสโดยใช้หม้อแปลงกระแส

ที่มา: http://www.thaieasyelec.com/article-wiki/review-product-article/



รูปที่ 2.8 สัญลักษณ์หม้อแปลงกระแส

ที่มา: http://www.thaieasyelec.com/article-wiki/review-product-article/

หม้อแปลงกระแสเป็นการวัดกระแสไฟฟ้าทางอ้อม โดยใช้หลักการเหนี่ยวนำของสนามแม่เหล็กเหมือนกับหม้อแปลงไฟฟ้า แต่เปลี่ยนให้ฝั่งต้นทาง (Primary) เป็นสายไฟที่ต้องการวัดกระแสแทนและมีเพียงขดลวดฝั่งปลายทาง (Secondary) เรียกว่าหม้อแปลงกระแส (Current Transformer) เมื่อเราจ่ายกระแสไฟฟ้าสลับไหลผ่านสายไฟ จะทำให้เกิดสนามแม่เหล็กเปลี่ยนแปลงไปมาและไปตัดกับขดลวดที่พันรอบแกนตัวตรวจรู้การเหนี่ยวนำ (Inductive Sensor) ทำให้เกิดกระแสไฟฟ้า

ขึ้นเมื่อต่อกับโหลด ซึ่งจะวัดได้เฉพาะกระแสไฟฟ้ากระแสสลับเท่านั้น กรณีที่จ่ายกระแสไฟฟ้ากระแสตรงเข้าไปในสายไฟ จะไม่มีการเปลี่ยนแปลงของสนามแม่เหล็ก ซึ่งจะไม่เกิดการเหนี่ยวนำของสนามแม่เหล็ก หลักการนี้สามารถนำไปใช้กับแคลมป์มิเตอร์ (Clamp meter)

รูปที่ 2.9 หม้อแปลงกระแสแบบต่าง ๆ
ที่มา: http://www.thaieasyelec.com/article-wiki/review-product-article/

ในกรณีที่เราวัดกระแสไฟฟ้าที่ฝั่งปลายทางได้ต่ำมาก เป็นเพราะโหลดกินกระแสน้อยมาก เมื่อเทียบกับย่านที่ตัวตรวจรู้วัดได้ ผู้ใช้ต้องเพิ่มสัญญาณให้มากกว่านี้ วิธีแก้ไขหนึ่งคือ การเพิ่มรอบขดลวดในฝั่งต้นทางเพื่อลดอัตราส่วนจำนวนรอบลง

2.1.3 ตัวตรวจรู้ PZEM-004T

รูปที่ 2.10 ตัวตรวจรู้ PZEM-004T

ที่มา: http://picclick.ca/AC-Multifunction-Meter-Watt-Power-Volt-Amp-Current-112062921932.html

2.1.3.1 ฟังก์ชัน

(1) ฟังก์ชันการวัดพารามิเตอร์ทางไฟฟ้า ได้แก่ แรงดันไฟฟ้า กระแสไฟฟ้า กำลังจริง พลังงาน

(2) ฟังก์ชันการตั้งค่าของคีย์พลังงาน

(3) เก็บข้อมูลได้เมื่อไม่มีไฟฟ้า เนื่องจากสามารถเก็บสะสมพลังงาน ก่อนที่จะหยุดการจ่ายไฟ

(4) ฟังก์ชันการแสดงผลที่เครื่องคอมพิวเตอร์ โดยสามารถแสดงผล แรงดันไฟฟ้า กระแสไฟฟ้า กำลังจริง พลังงาน

(5) ฟังก์ชันการสื่อสารแบบอนุกรม ด้วยอินเตอร์เฟซอนุกรมระดับแรงดันอยู่ที่ 0-5 โวลต์ (TTL) ของตัวเอง มีความหลากหลายของขั้วที่ใช้ในการสื่อสารที่สามารถเชื่อมต่อไปยังขาของบอร์ดและสามารถอ่านและตั้งค่าพารามิเตอร์ได้

2.1.3.2 การสื่อสารแบบอนุกรม

มอดูลนี้เป็นอุปกรณ์ที่มีทีทีแอล (TTL) ติดต่อสื่อสารข้อมูลแบบอนุกรม โดยสามารถอ่านและตั้งค่าพารามิเตอร์ที่เกี่ยวข้องผ่านทางพอร์ตอนุกรม แต่ถ้าต้องการที่จะสื่อสารที่มียูเอสบี (USB) หรือ RS232 ยกตัวอย่างเช่น คอมพิวเตอร์ เป็นต้น จำเป็นจะต้องมีการติดตั้งกับขาทีทีแอล (TTL) ของบอร์ดที่แตกต่างกัน

ตารางที่ 2.1 โปรโตคอลการสื่อสารของมอดูล

ลำดับที่	ฟังก์ชัน	หัวข้อ	ข้อมูล 1-5	ผลรวม
1	แรงดันไฟฟ้า	B0	C0 A8 01 01 00 คอมพิวเตอร์ส่งคำขอเพื่ออ่านค่าแรงดันไฟฟ้า	1A
		A0	00 E6 02 00 00 เครื่องวัดตอบกลับค่าแรงดันไฟฟ้าออกมาเป็น 230.2 โวลต์	88
2	กระแสไฟฟ้า	B1	C0 A8 01 01 00 คอมพิวเตอร์ส่งคำขอเพื่ออ่านค่ากระแสไฟฟ้า	1B
		A1	00 11 20 00 00 เครื่องวัดตอบกลับค่ากระแสไฟฟ้าออกมาเป็น 17.32 แอมแปร์	D2
3	กำลังจริง	B2	C0 A8 01 01 00 คอมพิวเตอร์ส่งคำขอเพื่ออ่านค่ากำลังจริง	1C
		A2	08 98 00 00 00 เครื่องวัดตอบกลับค่ากำลังจริงออกมาเป็น 2200 วัตต์	42
4	อ่านค่าพลังงาน	B3	C0 A8 01 01 00 คอมพิวเตอร์ส่งคำขอเพื่ออ่านค่าพลังงาน	1D
		A3	01 86 9f 00 00 เครื่องวัดตอบกลับค่าพลังงานออกมาเป็น 99999 วัตต์ต่อชั่วโมง	C9
5	ตั้งค่าที่อยู่ของมอดูล	B4	C0 A8 01 01 00 คอมพิวเตอร์ส่งคำขอเพื่อตั้งค่าที่อยู่ โดยที่อยู่คือ 192.168.1.1	1E
		A4	00 00 00 00 00 เครื่องวัดตอบกลับมาว่าการตั้งค่าที่อยู่สำเร็จ	A4
6	ตั้งค่าขอบเขตการแจ้งเตือนกำลังงาน	B5	C0 A8 01 01 14 คอมพิวเตอร์ส่งคำขอเพื่อตั้งค่าขอบเขตการแจ้งเตือนกำลังงาน	33
		A5	00 00 00 00 00 เครื่องวัดตอบกลับมาว่าการตั้งค่าขอบเขตการแจ้งเตือนกำลังงานสำเร็จ	A5

(1) การตั้งค่าที่อยู่การสื่อสาร 192.168.1.1

ส่งคำสั่ง B4 C0 A8 01 01 00 1E

ตอบกลับข้อมูล A4 00 00 00 00 00 A4

ตัวอย่างข้างต้น เป็นการแสดงให้เห็นว่าเป็นการตั้งที่อยู่ของการสื่อสาร ที่ 192.168.1.1 โดยที่ผู้ใช้สามารถตั้งค่าที่อยู่ตามความต้องการใช้งานได้ การส่งคำสั่งและการตอบกลับโดยอัตโนมัติที่แสดงไว้ข้างต้นเป็นข้อมูลที่แสดงในเลขฐาน 16 โดยที่ไบต์สุดท้ายของการส่งข้อมูลและการตอบกลับเป็น 1E และ A4 ตามลำดับ เป็นผลรวมสะสม ที่การส่งคำสั่ง B4+C0+A8+01+01+00=21E ใช้ผลรวมสะสมเลขฐาน 16 ของข้อมูลที่มีค่าเป็น 21E นำ 2 ไบต์สุดท้ายคือ 1E ใช้ในการส่งคำสั่ง ข้อมูลในการตอบกลับ A4+00+00+00+00+00+00=A4 ใช้ผลรวมสะสมเลขฐาน 16 ของข้อมูลที่มีค่าเป็นเป็น A4 ในการตอบกลับ

(2) การตั้งค่าขอบเขตการแจ้งเตือนกำลังงานที่ 20 กิโลวัตต์

ส่งคำสั่ง B5 C0 A8 01 01 14 33

ตอบกลับข้อมูล A5 00 00 00 00 00 A5

จากตัวเลข 14 ในการส่งคำสั่งเป็นค่าการแจ้งเตือน โดยที่ 14 เป็นข้อมูลที่แสดงในเลขฐาน 16 เมื่อแปลงจากเลขฐาน 16 มาเป็นเลขฐาน 10 จะมีค่าเป็น 20 เมื่อค่าการแจ้งเตือนกำลังงานของมอดูลนี้มีหน่วยเป็นกิโลวัตต์ โดยมีค่าน้อยที่สุดของการแจ้งเตือนอยู่ที่ 1 กิโลวัตต์ และค่ามากที่สุดของการแจ้งเตือนอยู่ที่ 22 กิโลวัตต์

(3) การอ่านค่าแรงดันไฟฟ้า

ส่งคำสั่ง B0 C0 A8 01 01 00 1A

ตอบกลับข้อมูล A0 00 E6 02 00 00 88

การตอบกลับข้อมูลแรงดันไฟฟ้าคือ D1D2D3=00 E6 02 โดยที่ 00 E6 แสดงถึงจำนวนเต็มของแรงดันไฟฟ้า และ 02 แสดงถึงเลขของแรงดันไฟฟ้าที่มีทศนิยม 1 ตำแหน่ง เมื่อแปลง 00 E6 ให้อยู่ในรูปของเลขฐาน 10 จะมีค่าเป็น 230 และเมื่อแปลง 02 ให้อยู่ในรูปเลขฐาน 10 จะมีค่าเป็น 2 ดังนั้นแรงดันไฟฟ้ามีค่าเป็น 230.2 โวลต์

(4) การอ่านค่ากระแสไฟฟ้า

ส่งคำสั่ง B1 C0 A8 01 01 00 1B

ตอบกลับข้อมูล A1 00 11 20 00 00 D2

การตอบกลับข้อมูลกระแสไฟฟ้าคือ D2D3=11 20 โดยที่ 11 แสดงถึงจำนวนเต็มของกระแสไฟฟ้า และ 20 แสดงถึงเลขของกระแสไฟฟ้าที่มีทศนิยม 2 ตำแหน่ง เมื่อแปลง 11 ให้อยู่ในรูปของเลขฐาน 10 จะมีค่าเป็น 17 และเมื่อแปลง 20 ให้อยู่ในรูปเลขฐาน 10 จะมีค่าเป็น 32 ดังนั้นกระแสไฟฟ้ามีค่าเป็น 17.32 แอมแปร์

(5) การอ่านค่ากำลังจริง

ส่งคำสั่ง B2 C0 A8 01 01 00 1C

ตอบกลับข้อมูล A2 08 98 00 00 00 42

การตอบกลับข้อมูลกำลังคือ D1D2=08 98 ทำการแปลง 08 98 ให้เป็นเลขฐาน 10 จะมีค่าเป็น 2200 ดังนั้นค่ากำลังจริงมีค่าเป็น 2200 วัตต์

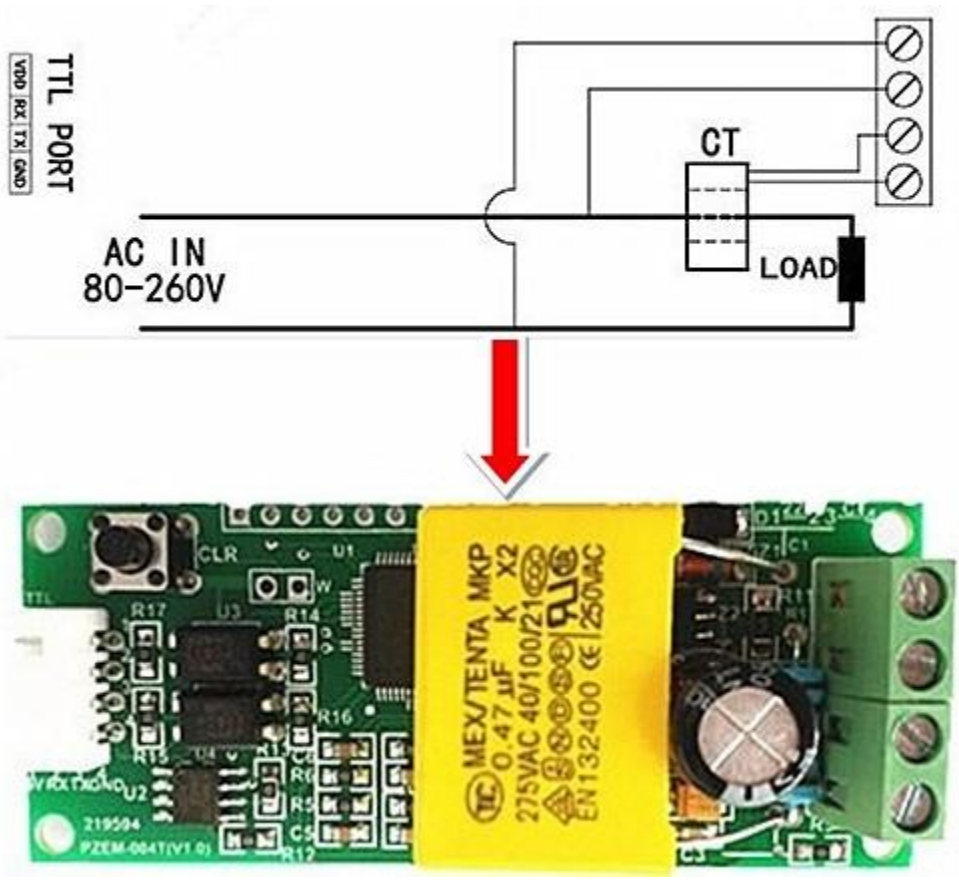
(6) อ่านค่าพลังงาน

ส่งคำสั่ง B3 C0 A8 01 01 00 1D

ตอบกลับข้อมูล A3 01 86 9F 00 C9

การตอบกลับข้อมูลพลังงานคือ D1D2D3=01 86 9F ทำการแปลง 01 86 9F ให้เป็นเลขฐาน 10 จะมีค่าเป็น 99999 ดังนั้นกำลังงานสะสมมีค่าเป็น 99999 วัตต์ต่อชั่วโมง

2.1.3.3 ภาพอธิบายการสื่อสาร



รูปที่ **2.11** แผนภาพการเดินสายไฟ **(**Wiring diagram**)**

ที่มา: https://ru.aliexpress.com/

(1) เชื่อมต่อสายตามแผนภาพการเดินสายไฟ

(2) หลังจากการต่อสายเป็นที่เรียบร้อย ให้ทำการเลือกพอร์ตของการสื่อสาร โดยที่ซอฟต์แวร์คอมพิวเตอร์พอร์ตที่สนับสนุนการสื่อสารของมอดูลนี้ประกอบไปด้วย COM2 COM3 COM4 โดยที่สามารถตรวจสอบผ่านการจัดการอุปกรณ์ ถ้าไม่ได้เป็นพอร์ตการสื่อสารข้างต้นให้ทำการแก้ไข

2.1.3.4 ข้อควรระวัง

(1) มอดูลนี้เหมาะสำหรับใช้ภายในอาคาร กรุณาอย่านำไปใช้ภายนอก

(2) โหลดที่ใช้ไม่ควรมีค่าเกินกำลังงานที่กำหนด

(3) ไม่ควรลำดับการเดินสายไฟผิด

2.1.3.5 รายละเอียดของพารามิเตอร์

(1) ทำงานที่แรงดันไฟฟ้า 80-260 โวลต์ของไฟฟ้ากระแสสลับ

(2) กำลังงานที่กำหนด คือ 100 แอมแปร์ต่อ 22000 วัตต์

(3) ความถี่ในการปฏิบัติงานอยู่ที่ 45-65 เฮิร์ต

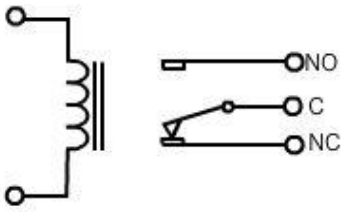
(4) ความแม่นยำของการทำการวัด อยู่ที่ระดับ 1.0

2.1.4 รีเลย์ (Relay)

รีเลย์ (Relay) คือ อุปกรณ์อิเล็กทรอนิกส์ที่ทำหน้าที่ ตัด ต่อ วงจร คล้ายกับสวิตช์ โดยใช้หลักการหน้าสัมผัส และการที่จะทำให้มันทำงานก็ต้องจ่ายไฟให้มันตามที่กำหนด เพราะเมื่อจ่ายไฟให้กับตัวรีเลย์ มันจะทำให้หน้าสัมผัสติดกัน กลายเป็นวงจรปิด และตรงกันข้ามทันทีที่ไม่ได้จ่ายไฟให้มัน มันก็จะกลายเป็นวงจรเปิด ไฟที่เราใช้ป้อนให้กับตัวรีเลย์ก็จะเป็นไฟที่มาจากอุปกรณ์ที่รีเลย์ทำการเชื่อมต่อ ดังนั้นทันทีที่เปิดอุปกรณ์เชื่อมต่อก็จะทำให้รีเลย์ทำงาน

รูปที่ 2.12 รีเลย์ที่ใช้งาน

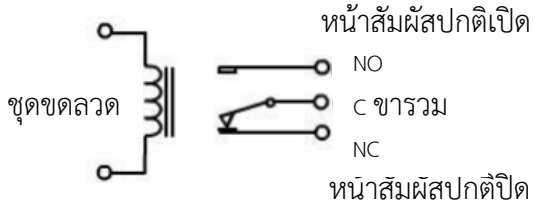
ที่มา: มงคล พรหมเทศ. งานไฟฟ้าทั่วไป. กรุงเทพมหานคร: เอมพันธุ์; 2542.



รูปที่ 2.13 สัญลักษณ์ของรีเลย์
ที่มา: มงคล พรหมเทศ. งานไฟฟ้าทั่วไป. กรุงเทพมหานคร: เอมพันธุ์; 2542.

2.1.4.1 โครงสร้างของรีเลย์

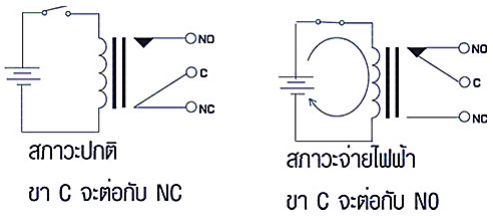
ภายในโครงสร้างของรีเลย์ จะประกอบไปด้วยขดลวด 1 ชุด และหน้าสัมผัส ซึ่งในหน้าสัมผัส 1 ชุด จะประกอบไปด้วย หน้าสัมผัสแบบปกติปิด (Normally Close หรือ NC) ซึ่งในสภาวะปกติ ขานี้จะต่ออยู่กับขาร่วม (C) และหน้าสัมผัสปกติเปิด (Normally Open หรือ NO) ขานี้จะต่อเข้ากับขาร่วม (C) เมื่อขดลวดมีแรงดันตกคร่อม หรือกระแสไหลผ่านในปริมาณที่เพียงพอ ในรีเลย์ 1 ตัว อาจมีหน้าสัมผัสมากกว่า 1 ชุด ซึ่งขึ้นอยู่กับผู้ผลิต



รูปที่ 2.14 สัญลักษณ์ของรีเลย์แทนโครงสร้างรีเลย์
ที่มา: มงคล พรหมเทศ. งานไฟฟ้าทั่วไป. กรุงเทพมหานคร: เอมพันธุ์; 2542.

2.1.4.2 หลักการทำงานของรีเลย์

เมื่อมีกระแสไฟฟ้าไหลผ่านขดลวด จะทำให้ขดลวดเกิดสนามแม่เหล็กไปดึง แผ่นหน้าสัมผัสลงมา แตะหน้าสัมผัสอีกอันทำให้มีกระแสไหลผ่านหน้าสัมผัสไปได้



รูปที่ 2.15 สภาวะการทำงานของรีเลย์
ที่มา: มงคล พรหมเทศ. งานไฟฟ้าทั่วไป. กรุงเทพมหานคร: เอมพันธุ์; 2542.

2.1.4.3 ข้อคำนึงถึงการใช้งานรีเลย์ทั่วไป

(1) แรงดันใช้งาน

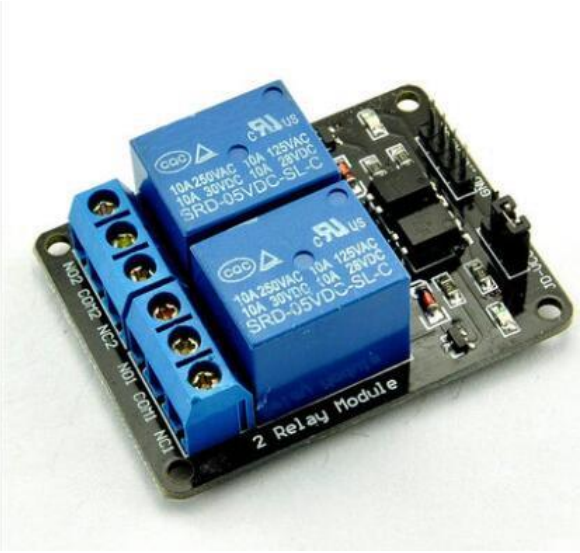
คือแรงดันที่ทำให้รีเลย์ทำงานได้ หากเราดูที่ตัวรีเลย์จะระบุค่า แรงดันที่ใช้งานเอาไว้ ซึ่งถ้าหากใช้แรงดันมากกว่าที่ระบุไว้ ขดลวดภายในตัวรีเลย์อาจจะขาดได้ หรือหากใช้แรงดันต่ำกว่ามาก รีเลย์จะไม่ทำงาน ส่วนในการต่อวงจรนั้นสามารถต่อขั้วใดก็ได้ เพราะตัวรีเลย์จะไม่ระบุขั้วต่อไว้ นอกจากรีเลย์ชนิดพิเศษ

(2) การใช้กระแสผ่านหน้าสัมผัส

ยกตัวอย่างเช่นตัวรีเลย์ระบุไว้ 10A 250VAC คือ หน้าสัมผัสของรีเลย์นั้นสามารถทนกระแสได้ 10 แอมแปร์ ที่ 250 โวลต์ ไฟฟ้ากระแสสลับ แต่การใช้งานควรที่จะใช้งานที่ระดับกระแสต่ำกว่าที่ระบุไว้ เพราะถ้ากระแสมากหน้าสัมผัสของรีเลย์จะเสียหายได้

2.1.4.4 มอดูลรีเลย์แบบ 2 ช่อง

บอร์ดรีเลย์ขนาด 2 ช่อง มีตัวต่อเอาต์พุตที่รีเลย์เป็น NO COM NC สามารถใช้ขับโหลดได้ทั้งแรงดันไฟฟ้ากระแสตรง (DC) และกระแสสลับ (AC) โดยใช้สัญญาณในการควบคุมการทำงานด้วยสัญญาณทรานซิสเตอร์ ทรานซิสเตอร์ลอจิก (TTL)



รูปที่ 2.16 บอร์ดรีเลย์ขนาด 2 ช่อง

ที่มา: http://www.thaieasyelec.com/products/components-th/2-channels-relay-module-detail.html

(1) คุณสมบัติ (Features)

(1.1) รีเลย์เอาต์พุตแบบมีหน้าสัมผัส 2 ชุดคือ หน้าสัมผัสแบบปกติเปิด (NO) และหน้าสัมผัสแบบปกติปิด (NC) เรียกว่า singpole-doublethrow (SPDT)

(1.2) สั่งงานด้วยระบบแรงดัน 0-5 โวลต์ (TTL)

(1.3) รับแรงดันได้สูงสุด 250 โวลต์ (กระแสสลับ) 10 แอมแปร์ หรือ 30 โวลต์ (กระแสตรง) 10 แอมแปร์

(1.4) มีแอลอีดี (LED) แสดงสถานการณ์ทำงานของรีเลย์และแสดงสถานะของบอร์ด

(1.5) มีจัมเปอร์สำหรับเลือกว่าจะใช้กราวด์ร่วมหรือแยก

(1.6) มี OPTO-ISOLATED เพื่อแยกกราวด์ส่วนของสัญญาณควบคุมกับไฟที่ขับรีเลย์ออกจากกัน

(2) คุณลักษณะ (Specification)

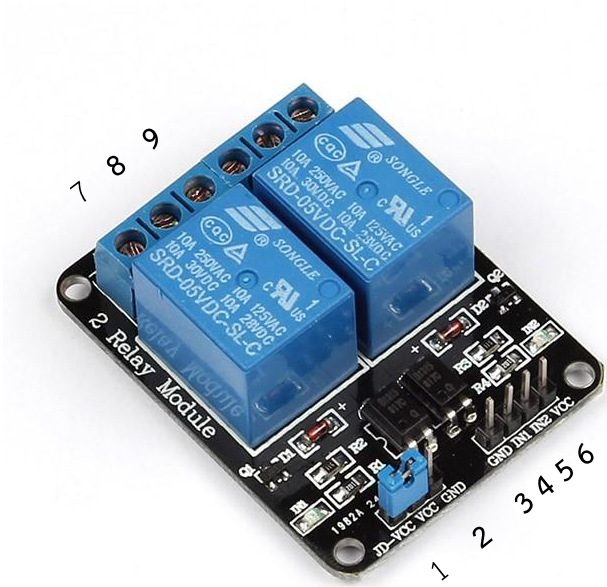
(2.1) ควบคุมไฟกระแสตรงได้สูงสุด 10 โวลต์ 10 แอมแปร์ และไฟกระแสสลับสูงสุด 250 โวลต์ 10 แอมแปร์

(2.2) ระดับสัญญาณอินพุตควบคุมแบบทรานซิสเตอร์ ทรานซิสเตอร์ลอจิก (TTL) ทำงานด้วยสัญญาณแบบแอกทีฟไฮ (Active High)

(2.3) ขนาดรูยึดบอร์ด 3 มิลลิเมตร

(2.4) ขนาด ยาว 55 มิลลิเมตร กว้าง 40 มิลลิเมตร และสูง 20 มิลลิเมตร

(3) ขาสัญญาณและการเชื่อมต่อ (Pin Definition and Rating)



รูปที่ 2.17 ขาสัญญาณ

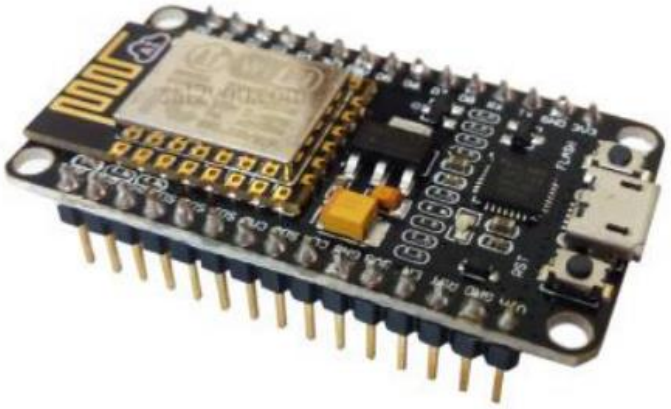
ตารางที่ 2.2 ขาสัญญาณและการเชื่อมต่อ

ขาที่	คำอธิบาย
1	+VCC ขาไฟ 5 โวลต์กระแสตรงจากแหล่งจ่ายภายนอก
2	GND กราวด์ของบอร์ด
3	GND กราวด์ของรีเลย์
4	ขาสัญญาณอินพุตรีเลย์ 1
5	ขาสัญญาณอินพุตรีเลย์ 2
6	+VCC ขาไฟ 5 โวลต์กระแสตรงจากบอร์ดอาร์ดูโน่
7	NC (Normally Closed) ซึ่งหมายถึงหน้าสัมผัสแบบปกติปิด
8	COM (Common) ที่จะตัดหรือต่อวงจรจากขาหน้าสัมผัสแบบปกติปิด (NC) หรือขาหน้าสัมผัสปกติเปิด (NO)
9	NO (Normally Opened) ซึ่งหมายถึงหน้าสัมผัสแบบปกติเปิด

2.1.5 NodeMCU

NodeMCU คือ แพลตฟอร์มหนึ่งที่ช่วยในการสร้างโปรเจคอินเตอร์เน็ตของทุกสรรพสิ่ง ที่ประกอบไปด้วยตัวบอร์ด (Development Kit) และซอฟต์แวร์บนบอร์ด (Firmware) ที่เป็นโอเพ่นซอร์ส (Open source) สามารถเขียนด้วยภาษา Lau ได้ ทำให้การใช้งานได้ง่ายขึ้น มาพร้อมกับมอดูลวายฟาย (WIFI ESP8266) ซึ่งเป็นหัวใจสำคัญในการใช้เชื่อมต่ออินเตอร์เน็ต ตัวมอดูล ESP8266 นั้นมีอยู่ด้วยกันหลายรุ่นมาก ตั้งแต่เวอร์ชั่น ESP-01 ไล่ไปเรื่อยๆ จนถึง ESP-12 และที่ฝังอยู่ใน NodeMCU เวอร์ชั่นแรกนั้นเป็น ESP-12 แต่ในเวอร์ชั่น 2 นั้นจะเป็น ESP-12E แทน ซึ่งการใช้งานโดยรวมก็ไม่ได้แตกต่างกันมาก นัก NodeMCU นั้นมีลักษณะคล้ายอาร์ดูโน่ (Arduino) ตรงที่มีพอร์ตอินพุตและเอาต์พุตมาในตัว สามารถเขียนโปรแกรมควบคุมอุปกรณ์ ได้โดยไม่ต้องผ่านอุปกรณ์อื่นๆ และเมื่อไม่นานมานี้ก็มีนักพัฒนาที่สามารถทำให้ Arduino IDE ใช้งานร่วมกับ NodeMCU ได้ จึงทำให้ใช้ภาษา C หรือ C++ ในการเขียนโปรแกรมได้ ทำให้สามารถใช้งานได้หลากหลายยิ่งขึ้น NodeMCU สามารถทำอะไรได้มากมาย โดยเฉพาะเรื่องที่เกี่ยวกับอินเตอร์เน็ตของทุกสรรพสิ่ง ไม่ว่าจะเป็นการทำเว็บขนาดเล็ก การควบคุมการเปิดปิดไฟผ่านวายฟายและอื่นๆ อีกมากมาย

2.1.5.1 NodeMCU Devkit 1.0 (ESP-12E) Version 2



รูปที่ 2.18 NodeMCU Devkit 1.0 (ESP-12E) Version 2
ที่มา: http://narong.ece.engr.tu.ac.th/ei444/document/ESP8266.pdf

2.1.5.2 คุณลักษณะของ NodeMCU

(1) ชุดพัฒนานี้ประกอบไปด้วยมอดูล ESP8266

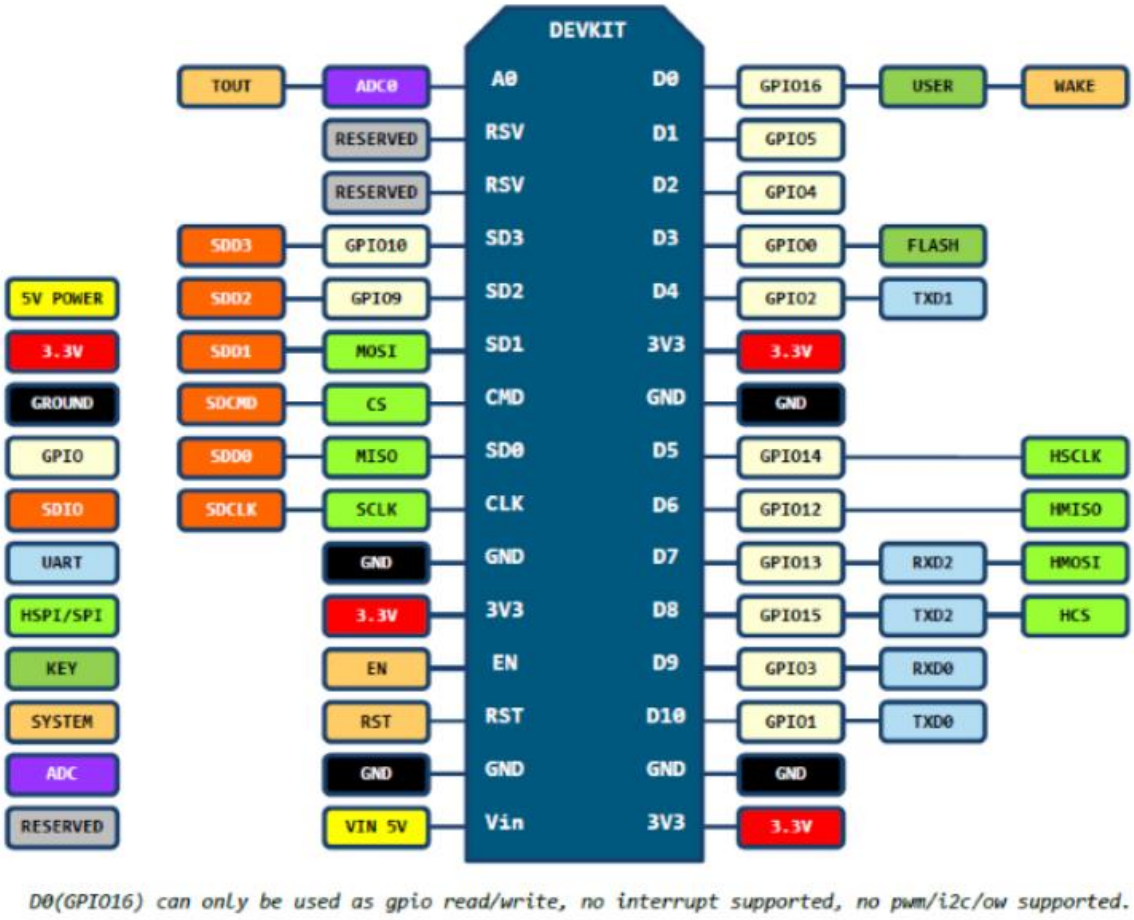
(2) มี GPIO PWM, 12C, 1-Wire และ ADC รวมอยู่บนบอร์ดเดียว

(3) มี USB-TTL มาในตัว ไม่จำเป็นต้องซื้อแยกเหมือนการใช้ ESP8266 ปกติ ทำให้ใช้งานได้สะดวกขึ้น

(4) มีขา GPIO 10 ขา ทุก ๆ ขาสามารถเป็น PWM, 12C และ 1-Wire ได้

(5) มี PCB Antenna สำหรับส่งสัญญาณไร้สาย

(6) ใช้คอนเนกเตอร์แบบไมโครยูเอสบี (Micro-USB) สำหรับจ่ายแรงดันไฟเลี้ยงหรือเท่ากับ +5 โวลต์ และสำหรับดาวน์โหลดเฟิร์มแวร์



รูปที่ 2.19 NodeMCU Devkit 1.0 (ESP-12E) Version 2
ที่มา: http://narong.ece.engr.tu.ac.th/ei444/document/ESP8266.pdf

2.1.6 การประมวลผลแบบกลุ่มเมฆ (Cloud Computing)

ปัจจุบันเทคโนโลยีการประมวลผลกลุ่มเมฆได้กลายเป็นหัวใจสำคัญของการบริหารจัดการต้นทุนด้านสารสนเทศของแต่ละหน่วยงาน ไล่ตั้งแต่บริษัทขนาดใหญ่ (Corporation) มาจนถึงผู้ประกอบการขนาดกลางและขนาดเล็ก (Small and Medium Enterprise: SME) อีกทั้งยังครอบคลุมไปถึงเหล่าผู้เริ่มต้นประกอบธุรกิจ (Star Up) ที่ยังไม่มีเงินถุงเงินถังสำหรับลงทุนด้านระบบสารสนเทศเป็นของตนเอง เรื่องราวของเทคโนโลยีการประมวลผลแบบกลุ่มเมฆมีผู้พูดถึงกันมากมาย และไม่เพียงเท่านั้น การเข้าถึงเพื่อขอใช้บริการระบบประมวลผลกลุ่มเมฆที่มีผู้ประกอบการให้บริการอยู่เป็นจำนวนมาก ทั้งในประเทศไทยและต่างประเทศก็ทำได้โดยสะดวก ยิ่งสำหรับผู้ใช้บริการที่เป็นผู้บริโภคโดยทั่วไปซึ่งปัจจุบันมีการใช้งานอุปกรณ์สมาร์ทโฟน (Smart Phone) และแทบเลต พีซี ในการใช้ชีวิตเพื่อกิจกรรมต่างๆ ก็มีแอพพลิเคชั่นซึ่งช่วยในการบริหารจัดการข้อมูลมากมาย ตั้งแต่การทำสำรอง (Backup) รูปภาพที่ถ่ายจากโทรศัพท์เคลื่อนที่ รายการข้อมูลต่างๆ รวมถึงข้อมูลมัลติมีเดียที่ใช้ประกอบไลฟ์สไตล์ของตนเองทั้งในแง่

ของการที่ให้ผู้บริโภคบริหารจัดการข้อมูลเอง อาทิ เช่น Dropbox หรือ Evemote ไปจนถึงแอพพลิเคชั่นใช้งาน เช่น Facebook, iTune, Instagram หรือแอพพลิเคชั่นให้บริการฟังเพลงอย่าง AIS Music Store KKBOX, DeeZer หรือ Spotify

รูปที่ 2.20 ส่วนหนึ่งของแอพพลิเคชั่นสำหรับผู้บริโภคที่ใช้ประโยชน์จากเทคโนโลยีการประมวลผลกลุ่มเมฆ

ที่มา: ไพโรจน์ ไววานิชกิจ. Cloud Computing หัวใจโลกธุรกิจยุคใหม่. เซมิคอนดักเตอร์ อิเล็กทรอนิกส์. 2558:101.

2.1.6.1 บริการพื้นฐานของการให้บริการการประมวลผลกลุ่มเมฆ

(1) บริการด้านโครงข่าย (Infrastructure)

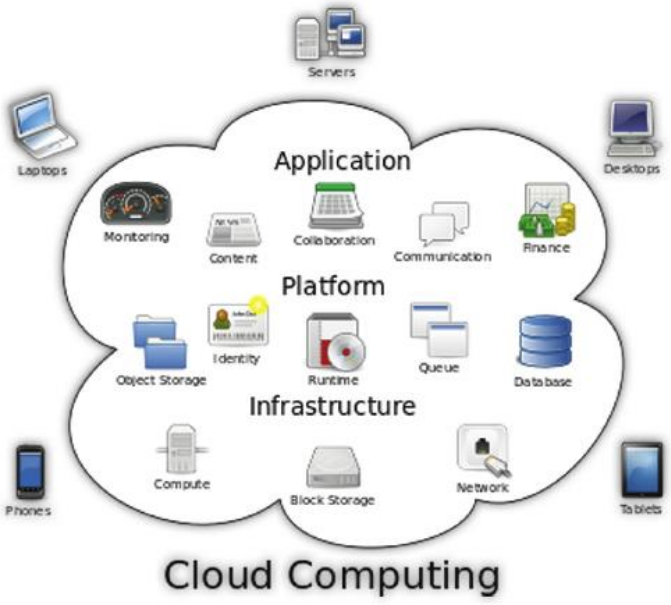
ประกอบไปด้วยความสามารถในการประมวลผล บริการเก็บข้อมูลของผู้ใช้งาน ทั้งในแง่ของผู้บริโภคและการใช้งานโดยผู้ประกอบการ โดยแบ่งแยกพื้นที่ในการจัดเก็บอย่างชัดเจน และการรับประกันความพร้อมของระบบเครือข่ายที่ใช้ในการเข้าถึงบริการด้านการประมวลผลกลุ่มเมฆ

(2) บริการด้านแพลตฟอร์ม (Platform)

ประกอบด้วยการให้บริการเก็บข้อมูลตามประเภทของการใช้งาน การให้บริการแยกแยะตัวตนของผู้ใช้บริการ (Identity) การให้บริการระบบประมวลผลแบบ Runtime การให้บริการจัดคิวการประมวลผลและการบริหารจัดการระบบฐานข้อมูล

(3) บริการด้านแอพพลิเคชั่น (Application)

ให้บริการเสริมสำหรับการสร้างแอพพลิเคชั่นบนเครือข่าย การประมวลผลกลุ่มเมฆ ประกอบด้วยการตรวจคุณภาพให้บริการและการใช้งาน (Monitoring) การบริหารจัดการเนื้อหาข้อมูล (Content) การให้บริการสำหรับการเข้าใช้งานร่วมกันระหว่างผู้ใช้บริการหลายๆ รายในแอพพลิเคชั่นเดียวกัน (Collaboration) บริการสนับสนุนการสื่อสารระหว่างผู้ใช้งาน และบริการด้านการคำนวณและจัดการด้านการเงินและค่าสวัสดิการ



รูปที่ 2.21 รายละเอียดของบริการต่างๆ ภายในเครือข่ายประมวลผลกลุ่มเมฆ
ที่มา: ไพโรจน์ ไววานิชกิจ. Cloud Computing หัวใจโลกธุรกิจยุคใหม่. เซมิคอนดักเตอร์ อิเล็กทรอนิกส์. 2558:103.

2.1.6.2 รูปแบบการให้บริการการประมวลผลกลุ่มเมฆ

การเลือกให้บริการการประมวลผลกลุ่มเมฆ สำหรับใช้งานของหน่วยงานธุรกิจต่างๆ นั้นมีสิ่งที่ต้องคำนึงถึงอยู่ 2 ประการก็คือรูปแบบของการใช้บริการ (Service Model) และรูปแบบของการติดตั้งใช้งาน (Deployment Model) ซึ่งหน่วยงานธุรกิจพึงทำความเข้าใจ

(1) รูปแบบการให้บริการชนิด Infrastructure as a Service

เป็นบริการที่ผู้ใช้บริการเลือกใช้เฉพาะบริการขั้นพื้นฐาน ของการประมวลผลกลุ่มเมฆ เปรียบได้กับการส่งถ่ายงานการลงทุนทางด้านฮาร์ดแวร์ของระบบสารสนเทศ จากการที่หน่วยงานธุรกิจต้องลงทุนเอง กลายเป็นไปเช่าใช้บริการฮาร์ดแวร์จากผู้ให้บริการการประมวลผลกลุ่มเมฆ แต่หน่วยงานธุรกิจที่เป็นผู้ใช้งานนี้ยังคงรักษาสิทธิ์ในการติดตั้งระบบปฏิบัติการและซอฟต์แวร์ รวมไปถึงการสร้างและบริหารจัดการแอพพลิเคชั่นใช้งานด้วยตนเอง ไม่ให้ผู้บริการประมวลผลกลุ่มเมฆเข้ามายุ่งเกี่ยว ในหลายๆ กรณีมักเรียกรูปแบบการเช่นใช้การประมวลผลกลุ่มเมฆ แบบ IaaS ว่าเป็นการใช้งานในระดับกายภาพ (Physical Layer)

การให้บริการแบบ IaaS ถือเป็นรูปแบบของการประมวลผลกลุ่มเมฆที่ง่ายที่สุด ซึ่งหน่วยงาน Internet Engineering Task Force (IETF) ซึ่งเป็นผู้วางมาตรฐานทางด้านเทคโนโลยีอินเตอร์เน็ตจากประเทศสหรัฐอเมริกาได้ให้คำนิยามบริการ IETF ว่าเป็นการที่ผู้ให้บริการประมวลผลกลุ่มเมฆ ลงทุนสร้างเครือข่ายเครื่องคอมพิวเตอร์ ไม่ว่าจะเป็นการจัดสรรกลุ่มของเครื่องคอมพิวเตอร์โดย

เฉพาะเจาะจงให้กับลูกค้าผู้ใช้บริการ หรือจะเป็นการทำ Virtualization กลุ่มเครื่องคอมพิวเตอร์ระบบเครือข่ายกลุ่มเมฆของตนเองเพื่อให้บริการแก่ลูกค้าของตนในลักษณะของเครื่องคอมพิวเตอร์เสมือน ซึ่งนอกจากเครื่องคอมพิวเตอร์ที่ใช้ในการประมวลผลแล้ว ยังรวมถึงทรัพยากรอื่นๆ บนเครือข่ายกลุ่มเมฆ เช่นอุปกรณ์บันทึกข้อมูล ซึ่งส่วนใหญ่เป็นหน่วยบันทึกข้อมูลเสมือน นั่นคือไม่ได้มีการแปลงฮาร์ดดิสก์ทางกายภาพแบบจำเพาะเจาะจงให้กับลูกค้า หากแต่เป็นการกันพื้นที่ในเชิงตรรกะซึ่งอาจกระจายไปบนฮาร์ดดิสก์หลายๆ ลูก เพื่อจัดสรรให้กับผู้ขอใช้บริการประมวลผลแบบกลุ่มเมฆแบบ laaS นอกจากนั้นบริการแบบ laaS ยังครอบคลุมรวมไปถึงการที่ผู้ให้บริการเครือข่ายประมวลผลกลุ่มเมฆ ให้บริการอื่นๆ เช่น Firewall, Load Balancer เพื่อแบ่งและจัดสรรปริมาณข้อมูลที่ถูกส่งมาบนคอมพิวเตอร์บนเครือข่ายกลุ่มเมฆให้สมดุลกัน การจัดสรรเลขหมาย IP ไปจนถึงการสร้างเครือข่ายเชื่อมต่อแบบ Virtual LAN จากเครือข่ายกลุ่มเมฆไปยังสำนักงานของหน่วยงานธุรกิจที่ใช้บริการ laaS ผ่านเครือข่ายอินเตอร์เน็ตสาธารณะ อันเป็นการรักษาความปลอดภัยของข้อมูลที่มีการรับส่งระหว่างหน่วยงานธุรกิจกับผู้ให้บริการประมวลผลกลุ่มเมฆ

การให้บริการทรัพยากรเหล่านี้มีเงื่อนไขแบบ On-demand นั่นคือผู้ใช้บริการประมวลผลกลุ่มเมฆ ไม่ได้มีการจัดเตรียมขนาดของทรัพยากรเหล่านี้ไว้ตายตัวสำหรับลูกค้าแต่ละราย แต่จะทำการเพิ่มขนาดของทรัพยากร เช่น ขนาดของพื้นที่จัดเก็บข้อมูล ความจุของวงจรเชื่อมต่อแบบ Virtual Local Area Network (VLAN) ให้เป็นไปตามปริมาณการใช้งานของลูกค้าแต่ละราย ซึ่งหน่วยงานธุรกิจสามารถเลือกได้ว่าจะใช้การเชื่อมต่อจากเครือข่ายสารสนเทศภายในองค์กรของตน เข้าสู่เครือข่ายการประมวลผลแบบกลุ่มเมฆของผู้ให้บริการเป็นการเชื่อมต่อผ่านเครือข่ายอินเตอร์เน็ต แล้วใช้เทคโนโลยี VLAN รักษาความปลอดภัยของข้อมูล หรือเป็นการเช่าวงจรเชื่อมต่อผ่านเครือข่ายการประมวลผลกลุ่มเมฆ ที่ผู้ให้บริการกลุ่มเมฆอาจมีการเตรียมไว้หรือผ่านผู้ให้บริการเครือข่ายโทรคมนาคม

ในการดำเนินธุรกิจของหน่วยงานธุรกิจ ก็มองเพียงว่าองค์กรของตนได้ทำการส่งต่องาน (Outsource) การลงทุนด้านฮาร์ดแวร์ไปให้กับผู้ให้บริการประมวลผลกลุ่มเมฆแล้ว หน้าที่รับผิดชอบของหน่วยงานธุรกิจจะยังคงอยู่ในเรื่องของการลงทุนและบริหารจัดการระบบปฏิบัติการ ซอฟต์แวร์ และการพัฒนาแอพพลิเคชั่นในงานต่างๆ ซึ่งนั่นย่อมหมายความว่าหน่วยงานธุรกิจยังมีหน้าที่ในการรับผิดชอบบำรุงรักษาทางด้านซอฟต์แวร์และแอพพลิเคชั่น กล่าวได้ว่าการใช้บริการประมวลผลกลุ่มเมฆแบบ laaS นั้นเป็นเสมือนการแบ่งงานด้านการลงทุน เฉพาะฮาร์ดแวร์ของระบบสารสนเทศให้กับผู้ให้บริการประมวลผลแบบกลุ่มเมฆเท่านั้น เงินค่าใช้จ่ายของการใช้บริการแบบ laaS จะถูกคิดคำนวณในรูปของทรัพยากรเครือข่ายประมวลผลกลุ่มเมฆที่ผู้ให้บริการได้ทำการจัดสรรให้กับหน่วยธุรกิจ

(2) รูปแบบการให้บริการชนิด Platform as a Service

ขอบเขตหน้าที่ของผู้ให้บริการประมวลผลแบบกลุ่มเมฆจะเพิ่มมากขึ้นกว่าในการให้เช่าบริการแบบ laaS โดยมีการเพิ่มหน้าที่ในการจัดเตรียมระบบปฏิบัติการ ซึ่งแน่นอนว่าภายในเครือข่ายการ

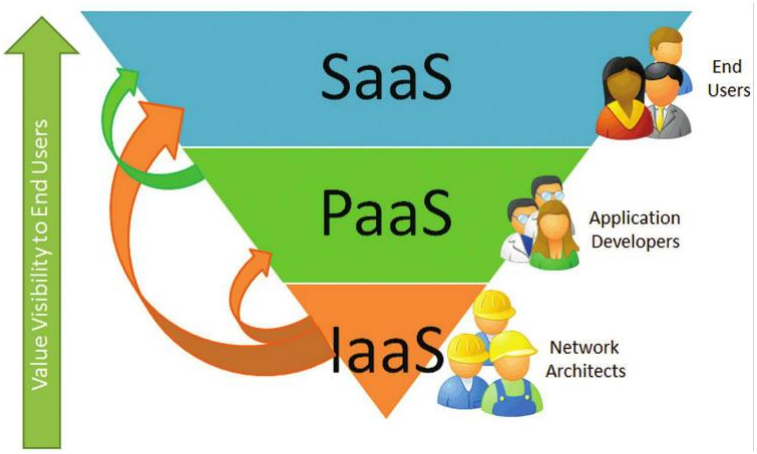
ประมวลผลแบบกลุ่มเมฆของผู้ให้บริการซึ่งประกอบไปด้วยเครื่องคอมพิวเตอร์จำนวนมหาศาล ก็ย่อมต้องมีการทำ Segmentation เพื่อแบ่งกลุ่มเครื่องคอมพิวเตอร์ที่มีการแบ่งหน้าที่แบบ Virtuallization ให้ประกอบไปด้วยกลุ่มที่มีระบบปฏิบัติการวินโดว์ (Windows) ระบบปฏิบัติการ UNIX ฯลฯ เพื่อเตรียมไว้ให้ผู้ใช้บริการที่สนใจเช่าบริการแบบ PaaS ได้เลือกใช้งาน และย่อมจะต้องมีกลุ่มเครื่องคอมพิวเตอร์แบบ Virtualization ที่ไม่ได้ลงระบบปฏิบัติการใด ๆ ไว้เพื่อให้บริการแบบ laaS สำหรับที่จะให้ผู้ใช้บริการนำไปลงระบบปฏิบัติการที่ตนเองต้องการควบคุมและถือครอง

นอกจากการจัดเตรียมระบบปฏิบัติการสำหรับการให้เช่าการประมวลผลแบบกลุ่มเมฆแบบ PaaS ผู้ให้บริการยังจะต้องมีการจัดเตรียม Program Language เช่น C, C++, Virtual C, HTML ฯลฯ ระบบประมวลผล (Execution Environment) ระบบฐานข้อมูลสำหรับการเขียนโปรแกรมและบันทึกข้อมูล รวมถึงบริการแบบเว็บเซิร์ฟเวอร์ (Web server) เพื่อให้ลูกค้าที่เป็นหน่วยงานธุรกิจที่ต้องการส่งต่องานด้านการบริการจัดการสิ่งเหล่านี้ นอกเหนือจากการหวังเช่าใช้เฉพาะฮาร์ดแวร์เสมือนในกรณีของ laaS การเช่าใช้บริการการประมวลผลแบบกลุ่มเมฆแบบ PaaS นี้มีข้อดีเหนือกว่า laaS ตรงที่สะดวกต่อการที่หน่วยงานธุรกิจจะมุ่งตรงไปที่การพัฒนาแอพพลิเคชั่น โดยไม่ต้องวุ่นวายกับการลงทุนและบริหารจัดการระบบปฏิบัติการและซอฟต์แวร์สนับสนุนอื่นๆ รวมทั้งยังสามารถเรียกใช้บริการเว็บเซิร์ฟเวอร์ จากทางผู้ให้บริการประมวลผลแบบกลุ่มเมฆได้โดยตรง ตัวอย่างของบริการแบบ PaaS จากผู้ให้บริการประมวลผลแบบกลุ่มเมฆระดับโลกก็ได้แก่ Microsoft Azure และ Google App Engine ซึ่งมีการบริหารจัดการระบบทรัพยากรตั้งแต่ระดับฮาร์ดแวร์และซอฟต์แวร์ ไปจนถึงอุปกรณ์บันทึกข้อมูลให้ความยืดหยุ่นทั้งในแง่ของขีดความสามารถในการประมวลผล และขนาดของหน่วยบันทึกข้อมูล เพื่อสอดรับกับปริมาณความต้องการใช้งานของหน่วยงานธุรกิจ เป็นการบริหารจัดการแบบอัตโนมัติ ทำให้หน่วยงานธุรกิจไม่จำเป็นต้องคอยติดต่อประสานงานกับบริษัท ผู้ให้บริการเพื่อขอปรับเปลี่ยนระดับของการให้บริการ ผลดีที่เกิดขึ้นก็คือทำให้หน่วยงานธุรกิจมีความมั่นใจว่าการเช่าใช้บริการการประมวลผลแบบกลุ่มเมฆแบบ PaaS จะช่วยทำให้ระบบสารสนเทศสามารถเติบโตได้อย่างไม่สะดุด ภายใต้หลักการประมวลผลแบบ Real-time Environment ภายใต้การเชื่อมต่อกับเครือข่ายประมวลผลแบบกลุ่มเมฆ

(3) รูปแบบการให้บริการชนิด Software as a Service

เป็นรูปแบบการให้บริการประมวลผลแบบกลุ่มเมฆแบบเบ็ดเสร็จ ซึ่งผู้ให้บริการกลุ่มเมฆจะทำหน้าที่บริหารจัดการทั้งระบบเครือข่ายที่เป็นฮาร์ดแวร์ รวมจนถึงแพลตฟอร์มที่เป็นระบบปฏิบัติการและการเขียนโปรแกรมต่างๆ เป็นพื้นฐาน แล้วต่อยอดให้บริการแอพพลิเคชั่นเพื่อให้ลูกค้าที่เป็นหน่วยงานธุรกิจสามารถเรียกใช้งานได้โดยไม่ต้องเสียเวลาทำการเขียนแอพพลิเคชั่นด้วยตนเอง ในหลายๆ กรณีมักมีการเรียกให้บริการแบบ SaaS ว่า On-demand Software โดยผู้ให้บริการการประมวลผลแบบกลุ่มเมฆมีการคิดราคาค่าบริการจากลูกค้าจากปริมาณการใช้งาน (Pay-per-use) และมักมีการกำหนดรูปแบบการ

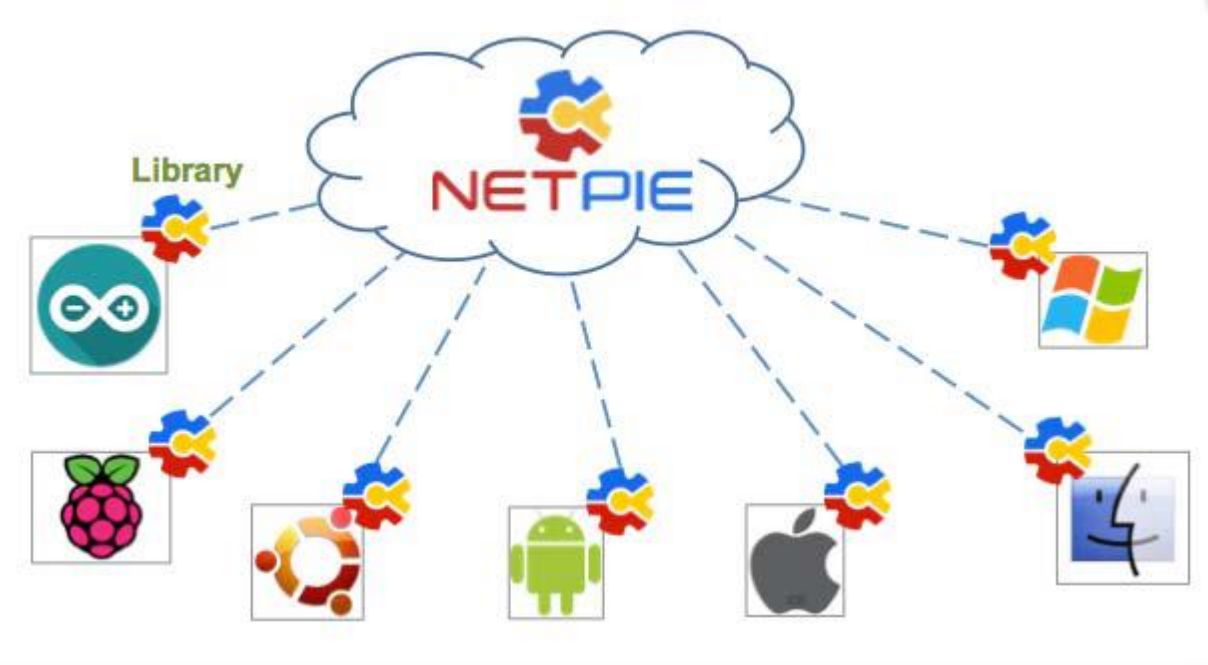
คิดเงินในลักษณะของการสมัครสมาชิกใช้งาน หากเทียบกับการใช้งานในลักษณะของผู้บริโภค ตัวอย่างของบริการชนิดนี้ก็ได้แก่บริการ Cloud Storage อย่างแอพพลิเคชั่น Dropbox นั่นเอง



รูปที่ 2.22 รูปแบบของการเลือกใช้บริการการประมวลผลแบบกลุ่มเมฆ
ที่มา: ไพโรจน์ ไววานิชกิจ. Cloud Computing หัวใจโลกธุรกิจยุคใหม่. เซมิคอนดักเตอร์ อิเล็กทรอนิกส์. 2558:105.

2.1.7 เน็ตพาย (NETPIE)

เน็ตพาย (NETPIE) เป็นอินเตอร์เน็ตของทุกสรรพสิ่ง Cloud Platform ที่พัฒนาขึ้นโดยทีมงานวิจัยและเปิดให้บุคคลทั่วไปใช้งานโดยมี Web portal ที่สามารถลงทะเบียนและจัดการตัวตนและสิทธิ์ของแอพพลิเคชั่นและอุปกรณ์ได้ที่ https://netpie.io ตั้งแต่เดือนกันยายน 2558 เป็นต้นมา เน็ตพายเป็น Middleware ที่มีหัวใจหลัก นอกเหนือจากส่วนอื่นๆ เป็น Distributed MQTT Brokers ซึ่งเป็นเสมือนจุดนัดพบให้สิ่งต่างๆ (Things) มาติดต่อสื่อสารและทำงานร่วมกันผ่านวิธีการส่งข้อความแบบ Publish/Subscribe เน็ตพายมีโครงสร้างสถาปัตยกรรมเป็นคลาวด์อย่างแท้จริงในทุกองค์ประกอบ ทำให้สามารถขยายตัวได้อย่างอัตโนมัติ (Auto-scale) สามารถดูแลและซ่อมแซมตัวเองได้อัตโนมัติเมื่อส่วนหนึ่งส่วนใดในระบบมีปัญหา (Self-healing, Self-recovery) โดยไม่ต้องพึ่งผู้ดูแลระบบ การบริหารจัดการระบบเป็นแบบ Plug-and-Play ไม่ต้องปรับแต่ง ในฝั่งอุปกรณ์เน็ตพายมี Client Library หรือที่เรียกว่าไมโครเกียร์ (MicroGear) ซึ่งทำหน้าที่สร้างและดูแลช่องทางการสื่อสารระหว่าอุปกรณ์กับเน็ตพาย รวมไปถึงรักษาความปลอดภัยในการส่งข้อมูล โดยไมโครเกียร์เป็นโอเพ่นซอร์ส (Open Source)



รูปที่ 2.23 วิธีการสื่อสารของสิ่งต่าง ๆ ผ่านเน็ตพาย
ที่มา: NECTEC a member of NSTDA. คู่มือการใช้งาน NETPIE. 2559.

2.1.7.1 ประโยชน์ของเน็ตพาย

(1) ช่วยลดการใช้ทรัพยากรของการเชื่อมต่อ

เน็ตพาย (NETPIE) ช่วยให้อุปกรณ์สามารถสื่อสารกันได้โดยผู้ใช้ไม่ต้องกังวลว่า อุปกรณ์นั้นจะอยู่ที่ใด เพียงแค่นำไมโครเกียร์ไลบรารี่ (NETPIE library) ไปติดตั้งในอุปกรณ์เน็ตพายจะรับหน้าที่ดูแลการเชื่อมต่อให้ทั้งหมด ไม่ว่าอุปกรณ์นั้นจะอยู่ในเครือข่ายชนิดใด ลักษณะใด หรือแม้กระทั่งเคลื่อนย้ายไปอยู่ที่ใด ผู้ใช้สามารถตัดปัญหากวนใจในการที่จะต้องมาออกแบบการเข้าถึงอุปกรณ์จากระยะไกล (Remote access) ด้วยวิธีแบบเดิมๆ เช่นการใช้ fixed public IP หรือการติดตั้ง port forwarding ในเราเตอร์ (router) หรือการต้องไปลงทะเบียนกับผู้ให้บริการ dynamic DNS ซึ่งทั้งหมดล้วนมีความยุ่งยากและลดความยืดหยุ่นของระบบไม่เพียงเท่านั้นเน็ตพาย ยังช่วยให้การเริ่มต้นใช้งานเป็นไปโดยง่ายโดยการออกแบบให้อุปกรณ์ถูกค้นพบและเข้าสู่บริการโดยอัตโนมัติ (Automatic discovery, plug and play)

(2) ช่วยลดภาระด้านความปลอดภัยของระบบ

เน็ตพายถูกออกแบบให้มีระดับและสิทธิในการเข้าถึงในระดับ fine gain กล่าวคือผู้ใช้สามารถออกแบบเองได้ทั้งหมด เช่น สิ่งใดมีสิทธิ์คุยกับสิ่งใด สิ่งใดมีสิทธิ์หรือไม่เพียงใดในการอ่านหรือเขียนข้อมูล และสิทธิเหล่านี้จะมีอายุเท่าใดหรือถูกเพิกถอนภายใต้เงื่อนไขใด เป็นต้น

(3) ยืดหยุ่นต่อการขยายระบบ

เน็ตพายมีสถาปัตยกรรมเป็นคลาวด์ (Cloud) อย่างแท้จริงในทุกองค์ประกอบของระบบ ทำให้เกิดความยืดหยุ่น และคล่องตัวสูงในการขยายตัว นอกจากนี้ มอดูลต่างๆ ยังถูกออกแบบให้ทำงานแยกจากกันเพื่อให้เกิดสภาวะ loose coupling และสื่อสารกันด้วยวิธีการ asynchronous messaging ช่วยให้แพลตฟอร์มมีความน่าเชื่อถือสูง สามารถนำไปใช้ซ้ำ และพัฒนาต่อเติมได้ง่าย ดังนั้นผู้พัฒนาไม่จำเป็นต้องกังวลกับการขยายตัวเพื่อรับโหลดที่เพิ่มขึ้นในระบบอีกต่อไป

2.1.7.2 จุดเด่นของเทคโนโลยี (Innovation Statement)

(1) ออกแบบเพื่อรองรับการขยายตัวของระบบได้อย่างไร้ขีดจำกัด

(2) ทุกองค์ประกอบของแพลตฟอร์มมีส่วนซ้ำสำรองทำให้ระบบมีความพร้อมใช้สูง

(3) มีระบบบริหารจัดการ Identity ของอุปกรณ์ที่ยืดหยุ่น

2.1.7.3 คุณสมบัติและอัตราค่าบริการ

(1) รองรับการเชื่อมต่ออุปกรณ์แบบ Plug-and-Play

(2) รองรับการเชื่อมต่อสื่อสารได้ทุกที่ ทุกเวลา

(3) รองรับการสื่อสารของอุปกรณ์จำนวนมาก

(4) รองรับการจัดการสิทธิ์และการยืนยันตัวตนของอุปกรณ์ที่ยืดหยุ่น

(5) มี Open-source Library ที่รองรับระบบปฏิบัติการและฮาร์ดแวร์ที่หลากหลาย

ตารางที่ 2.3 อัตราค่าบริการ

	ฟรี	**กิจการ**
อุปกรณ์ (Devices)	100	มากกว่า 100
แอพไอดี (AppID)	ไม่จำกัด	ไม่จำกัด
ข้อความ (Message)	ไม่จำกัด	ไม่จำกัด
อัตรา (Rate)	1 ข้อความต่อวินาที	20 ข้อความต่อวินาที
ราคา (Price)	0 บาท ตลอด (ลงทะเบียน)	ติดต่อ

2.1.7.4 ชนิดของคีย์ (Key)

เน็ตพาย (NETPIE) รองรับอุปกรณ์ที่หลายหลาย นอกจากอุปกรณ์กายภาพเช่น บอร์ดฮาร์ดแวร์ต่าง ๆ แล้ว ยังรองรับอุปกรณ์เสมือน เช่น แอพพลิเคชั่นต่างๆ ซึ่งรวมไปถึงแอพพลิเคชั่นที่เขียนด้วย HTML5 ที่สามารถรันได้บนเบราเซอร์ ในกรณีเช่นนี้ การเชื่อมต่อไม่สามารถเป็นแบบการคงอยู่ (Persistent) ได้ เนื่องจากต้องมีการปิดเปิดเบราเซอร์ตลอดเวลา ดังนั้นการจัดการคีย์ (Key) และ Token

มีประสิทธิภาพ หลีกเลี่ยงการสร้าง Token ทิ้งไว้โดยไม่ได้ใช้งาน เน็ตพายจึงออกแบบให้มีคีย์อยู่ 2 ชนิด คือ

(1) Device Key

เมื่ออุปกรณ์ติดตั้ง Device Key และร้องขอ Token มายังเน็ตพาย การได้รับการอนุญาตสิทธิ อุปกรณ์จะได้รับ Token ที่ใช้ได้ตลอดไป ไม่มีการหมดอายุ ดังนั้นตราบใดที่ไม่มีการเพิกถอนสิทธิ อุปกรณ์สามารถใช้ Token เดิม ไปได้ตลอดโดยไม่ต้องเข้าสู่กระบวนการ Authorization ใหม่ Device Key จึงเหมาะกับอุปกรณ์กายภาพที่สามารถรักษาช่องการเชื่อมต่อกับเน็ตพายไว้ได้โดยตลอด

(2) Session Key

อุปกรณ์ที่ได้รับการติดตั้ง Session Key จะได้รับ Token ที่ใช้ได้เพียงครั้งเดียว (One-time Token) หากอุปกรณ์ยกเลิกการเชื่อมต่อไป และต่อกลับมาใหม่ จะต้องมีการนำ Session Key ไปขอ Token ใหม่ทุกครั้ง Session Key จึงเหมาะกับการใช้งานแอพพลิเคชั่นที่รันบนเบราเซอร์ที่ต้องมีการปิดเปิดอยู่เรื่อยๆ เพราะหากใช้ Device Key ทุกครั้งที่เบราเซอร์ถูกเปิดขึ้นจะมีการร้องขอ Token ใหม่ และเมื่อเบราเซอร์ถูกปิดไป Token ที่ได้จะค้างอยู่เช่นนั้นโดยไม่ได้ใช้งานอีก ส่งผลให้ภายใต้ Key เดียวกันมี Token ทั้งที่ใช้งานและไม่ได้ใช้งานแล้วผูกอยู่เป็นจำนวนมาก สร้างความสับสนซับซ้อนที่ไม่จำเป็นให้กับผู้ใช้ จึงขอเน้นย้ำอย่างยิ่งว่าไม่ควรใช้ Device Key กับแอพพลิเคชั่นที่รันบนเบราเซอร์ HTML5 โดยเด็ดขาด

2.1.7.5 ไมโครเกียร์ (Microgear)

ไมโครเกียร์ (Microgear) คือซอฟต์แวร์ไลบรารี่ของเน็ตพาย (NETPIE) ที่ติดตั้งอยู่บนอุปกรณ์ที่ต้องการเชื่อมต่อสื่อสารผ่านคลาวด์ของเน็ตพายไมโครเกียร์ (NETPIE Microgear) เปรียบเสมือนตัวกลางและผู้ช่วยในการสร้างและดูแลการเชื่อมต่อ ให้มีความเสถียร ปลอดภัย ให้การสื่อสารแลกเปลี่ยนข้อมูลระหว่างอุปกรณ์เป็นไปได้อย่างราบรื่น บทบาทหน้าที่ของไมโครเกียร์สามารถแบ่งออกเป็น 4 ด้าน คือ

(1) ด้านการสื่อสาร (Communication)

ไมโครเกียร์จะเป็นผู้ช่วยในการสร้างการเชื่อมต่อ (Connection) ไปยังคลาวด์ของเน็ตพาย (NETPIE) และคอยตรวจสอบสถานะของการเชื่อมต่อ หากการเชื่อมต่อมีปัญหาไมโครเกียร์ (Microgear) สามารถช่วยเชื่อมต่อให้ใหม่เพื่อให้การสื่อสารเป็นไปได้อย่างราบรื่น นอกจากนี้ไมโครเกียร์ (Microgear) ยังช่วยอำนวยความสะดวก ในช่องทางการสื่อสารแบบเข้ารหัสในกรณีที่ผู้ใช้ต้องการ ส่วนการแลกเปลี่ยนข้อมูลระหว่างไมโครเกียร์ (Microgear) และคลาวด์ของเน็ตพาย (NETPIE) จะใช้โพรโทคอล MQTT ในการสื่อสาร

(2) ด้านการยืนยันตัวตน (Authentication)

ในขั้นตอนการสร้างการเชื่อมต่อไมโครเกียร์ (Microgear) จะช่วยยืนยันตัวตนของอุปกรณ์กับคลาวด์ของเน็ตพาย (NETPIE) โดยการพิสูจน์ตัวตน (Identity) ของอุปกรณ์จะใช้ข้อมูลประกอบกันสามส่วนคือ AppID, App Key และ Token

(3) ด้านการขออนุญาตสิทธิ (Authorization)

การขอนุญาติสิทธิในการสื่อสารจะเกิดขึ้นในขั้นตอนการสร้างการเชื่อมต่อ ควบคู่กับการยืนยันตัวตน คลาวด์ของเน็ตพาย (NETPIE) จะเป็นผู้ออกใบอนุญาต (Token) ที่ระบุว่าอุปกรณ์ตัวนี้สามารถสื่อสารได้กับอุปกรณ์ตัวใดบ้าง ในกรณีปกติอุปกรณ์ที่อยู่ภายใต้กลุ่ม AppID เดียวกันเท่านั้น จึงจะมีสิทธิสื่อสารกันได้

(4) ด้านการประสานงาน (Coordination)

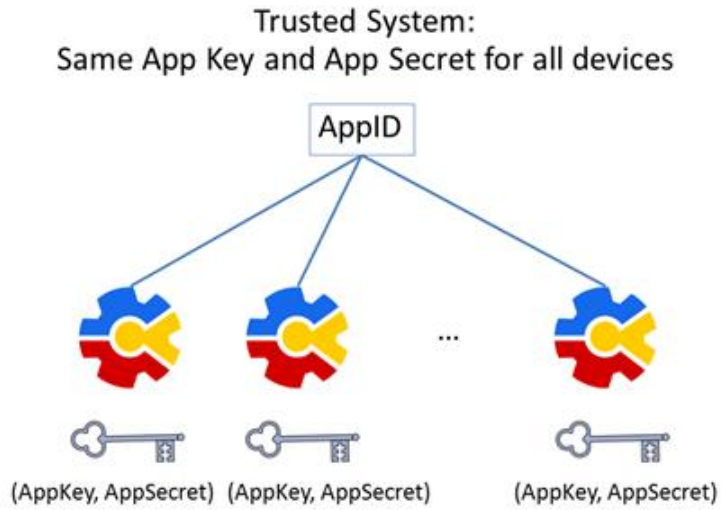
ไมโครเกียร์ (Microgear) มีฟังก์ชันที่ช่วยให้อุปกรณ์ต่างๆ ภายในกลุ่ม AppID เดียวกันทราบสถานะของกันและกัน เช่นทราบว่ามีอุปกรณ์ใดออนไลน์เข้ามาใหม่ในกลุ่ม หรือมีอุปกรณ์ใดออกไปจากกลุ่ม รวมถึงทราบการเปลี่ยนแปลงสถานะของอุปกรณ์ที่สนใจติดตาม จากข้อมูลดังกล่าวผู้ใช้สามารถกำหนดบทบาทหน้าที่ให้อุปกรณ์ในกลุ่มตามสถานะของอุปกรณ์อื่นๆ ในกลุ่ม เช่น หากเป็นอุปกรณ์ตัวแรกในกลุ่มให้ทำหน้าที่เป็นหัวหน้ากลุ่ม เป็นต้น

2.1.7.6 Key Management Concept

รูปแบบการจัดการสิทธิของแอพพลิเคชั่นและอุปกรณ์ เน็ตพาย (NETPIE) มีความยืดหยุ่นในการกำหนดสิทธิการเข้าถึงและการยืนยันตัวตนของแอพพลิเคชั่นและอุปกรณ์ โดยผู้ใช้เน็ตพาย (NETPIE) สามารถเลือกสร้างระบบได้ 3 แบบ ได้แก่

(1) แบบที่ 1 Trusted System

วิธีนี้เหมาะสำหรับ ระบบที่พัฒนาขึ้นมาเพื่อการทดสอบ หรือระบบที่อุปกรณ์ทุกตัวอยู่ภายใต้การดูแลของเจ้าของระบบเดียวกัน ในแบบนี้อุปกรณ์ทุกตัวในระบบจะถูกติดตั้ง App Key และ App Secret ชุดเดียวกันและมีสิทธิการเข้าถึงทุกอย่างเหมือนกันหมด ทำให้มีความสะดวกคล่องตัวสูงในการจัดการตัวตนและใช้งาน ตัวอย่างระบบที่เหมาะกับการใช้งานแบบนี้ ได้แก่ ระบบตัวตรวจรู้ (Sensor) สภาพแวดล้อมที่พัฒนาขึ้นเพื่อการใช้งานเฉพาะที่

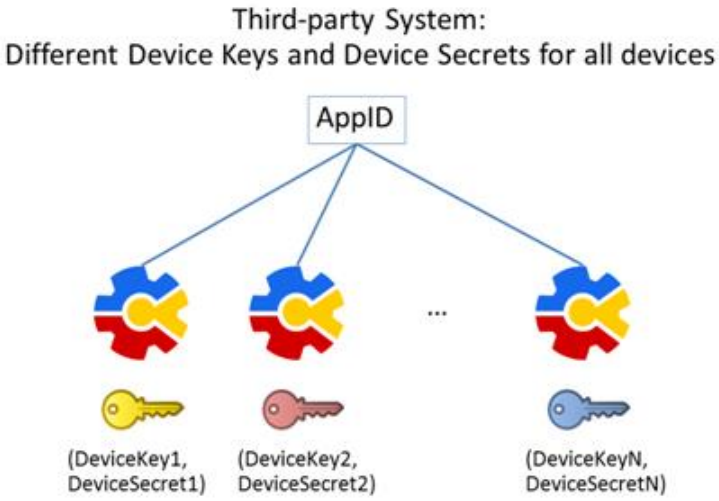


รูปที่ 2.24 Trusted System

ที่มา: NECTEC a member of NSTDA. คู่มือการใช้งาน NETPIE. 2559.

(2) แบบที่ 2 Third-party System

วิธีนี้เหมาะสำหรับ ระบบที่ให้บริการต่อบุคคลอื่น และ/หรือ เมื่ออุปกรณ์อยู่ในสภาพแวดล้อมที่นอกเหนือการควบคุมของเจ้าของระบบ อุปกรณ์ทุกตัวจะใช้ Device Key และ Device Secret ที่แตกต่างกันและมีสิทธิการเข้าถึง แตกต่างกันไป เนื่องจากอุปกรณ์แต่ละตัวมีการระบุตัวตนที่แตกต่างกัน วิธีการนี้มีความปลอดภัยสูง แต่ก็ทำให้การบริหารจัดการสิทธิและตัวตนอยากขึ้นกว่าแบบที่ 1 ตัวอย่างระบบที่เหมาะกับการใช้งานแบบนี้ ได้แก่ ระบบอุปกรณ์อำนวยความสะดวกในบ้าน เช่น หลอดไฟที่ควบคุมได้ผ่านแอพพลิเคชั่น เนื่องจากผู้พัฒนาไม่ได้พัฒนาหลอดไฟเพื่อใช้เองแต่เพื่อขายต่อ เมื่อผู้ใช้ซื้อหลอดไฟไปติดตั้งที่บ้านควรมีสิทธิในการควบคุมเฉพาะหลอดไฟในบ้านของตนเท่านั้น ดังนั้น Key ในหลอดไฟแต่ละดวงต้องแตกต่างกัน

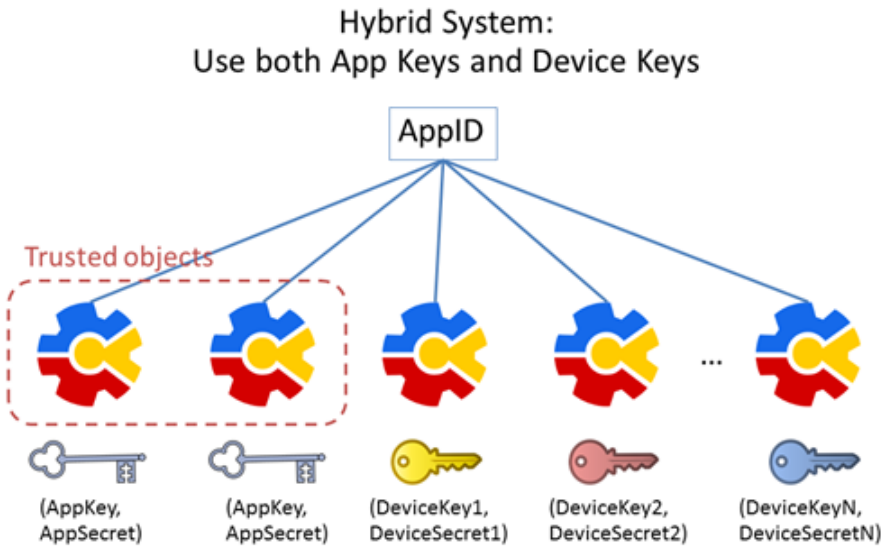


รูปที่ 2.25 Third-party System

ที่มา: NECTEC a member of NSTDA. คู่มือการใช้งาน NETPIE. 2559.

(3) แบบที่ 3 Hybrid System

วิธีนี้เหมาะสำหรับระบบที่ประกอบด้วยอุปกรณ์ของเจ้าของระบบเอง ซึ่งติดตั้งด้วย App Key, App Secret และอุปกรณ์ที่ใช้งานโดยบุคคลอื่นซึ่งติดตั้งด้วย Device Key, Device Secret เป็นการใช้งานที่ผสมผสานแบบ Trusted System และแบบ Third-party System



รูปที่ 2.26 Hybrid System

ที่มา: NECTEC a member of NSTDA. คู่มือการใช้งาน NETPIE. 2559.

2.1.7.7 อุปกรณ์ต่าง ๆ ที่ไมโครเกียร์ (Microgear) รองรับ

ตารางที่ 2.4 อุปกรณ์ต่าง ๆ ที่ไมโครเกียร์ (Microgear) รองรับและเน็ตพาย (NETPIE) สามารถประมวลผลได้

 รูปที่ 2.27 บอร์ดอาร์ดูโน่ (Arduino board)	อาร์ดูโน่ (Arduino) คือ แผงวงจรไมโครคอนโทรลเลอร์พื้นฐานที่นิยมใช้กัน - 8 bit microcontroller - Extremely popular - Simple to program - Extend functionality by integrating shields - Price varies between 2xx-1xxx bath
 รูปที่ 2.28 บอร์ดราสเบอร์รี่พาย (Raspberry Pi board)	ราสเบอร์รี่พาย (Raspberry Pi) แผงคอมพิวเตอร์ลินุกซ์ (Linux) ประกอบไปด้วย - Embedded Linux board - ARM Cortex-A7 900 MHz - 1 GB Ram - HDMI - USB port - Ethernet port - Price 20-35 USD
 รูปที่ 2.29 โทรศัพท์เคลื่อนที่	สิ่งของ (Things) ที่เป็นมือถือ (Mobile) รองรับระบบปฏิบัติการ IOS Device, Android

 รูปที่ 2.30 สิ่งของ (things)	สิ่งของ (Things) ที่เป็นระบบปฏิบัติการ Linux, Mac OS, Windows
 รูปที่ 2.31 ESP8266	ESP8266 แผงวงจรวายฟาย (Wi-Fi) และไมโครคอนโทรลเลอร์ที่สามารถโปรแกรมได้ - Wi-Fi module + MCU - 12C, SPI, PWM - Official firmware interfaces via serial AT command - Retail Price < 3 USD
 รูปที่ 2.32 NodeMCU	NodeMCU Dev Kit แผงวงจรที่ต่อยอดจาก ESP8266 - Based on ESP8266 + USB to serial module - Official firmware executes LUA script - Price < 9 USD
 รูปที่ 2.33 Spark Core	Spark Core สามารถโปรแกรมผ่านเว็บได้ - Microcontroller + Wi-Fi - ARM based Arduino environment - TI CC3000 Wi-Fi module support smart config - Program via web IDE - Compile and flash over the air via a free cloud service - Price 39 USD

 รูปที่ 2.34 สิ่งของ (Things)	สิ่งของ (Things) ที่เป็น Browser HTML5 - Pure HTML5 files running on a web browser without cooperation of web servers
 รูปที่ 2.35 ARM Embed	ARM Embed - STM32 Nucleo open development platform

2.1.8 แอนดรอยด์ (Android)

Android OS คือ ระบบปฏิบัติการสำหรับอุปกรณ์พกพา เช่น โทรศัพท์มือถือ แทบเลต คอมพิวเตอร์และเน็ตบุ๊ค ที่ทำงานบนลินุกซ์เคอร์เนล เริ่มพัฒนาโดยบริษัทแอนดรอยด์ จากนั้นบริษัทแอนดรอยด์ถูกซื้อโดยกูเกิล และทางกูเกิลได้นำแอนดรอยด์ไปพัฒนาต่อ ส่วนด้านลิขสิทธิ์ของโค้ดแอนดรอยด์จะใช้ในลักษณะของซอฟแวร์เสรีหรือโอเพ่นซอร์ส (Open Source) ทำให้นักพัฒนาสามารถแก้ไข ดัดแปลงโค้ดแอนดรอยด์ได้อย่างอิสระ และที่สำคัญคือแจกฟรี สำหรับระบบปฏิบัติการแอนดรอยด์เป็นที่รู้จักต่อสาธารณชน เมื่อวันที่ 5 พฤศจิกายน พ.ศ.2550 และแอนดรอยด์เวอร์ชั่น 1.0 ถูกปล่อยออกมาใช้งานอย่างเป็นทางการครั้งแรกเมื่อวันที่ 28 กันยายน พ.ศ.2551

2.1.8.1 โครงสร้างของแอนดรอยด์

การทำความเข้าใจโครงสร้างของระบบปฏิบัติการแอนดรอยด์ ถือว่าเป็นสิ่งสำคัญเพราะถ้านักพัฒนาโปรแกรมสามารถมองภาพโดยรวมของระบบได้ทั้งหมด จะทำให้สามารถเข้าใจถึงกระบวนการทำงานได้ดียิ่งขึ้น และสามารถนำไปช่วยในการออกแบบโปรแกรมที่ต้องการพัฒนา เพื่อให้เกิดประสิทธิภาพในการทำงาน โครงสร้างที่สำคัญต่างๆ ในระบบปฏิบัติการแอนดรอยด์มีดังนี้

(1) แอพพลิเคชั่น (Application)

ส่วนแอพพลิเคชั่นหรือส่วนของโปรแกรมที่มีมากับระบบปฏิบัติการ เป็นกลุ่มของโปรแกรมที่ผู้ใช้งานได้ทำการติดตั้งไว้ โดยผู้ใช้งานสามารถเรียกใช้โปรแกรมต่างๆ ได้โดยตรง ซึ่งการทำงานของแต่ละโปรแกรมจะเป็นไปตามที่ผู้พัฒนาโปรแกรมได้ออกแบบและเขียนโค้ดโปรแกรมเอาไว้

(2) Application Framework

เป็นส่วนที่มีการพัฒนาขึ้นเพื่อให้นักพัฒนาสามารถพัฒนาโปรแกรมได้สะดวก และมีประสิทธิภาพมากยิ่งขึ้น โดยนักพัฒนาไม่จำเป็นต้องพัฒนาในส่วนที่มีความยุ่งยากมากๆ เพียงแค่ทำการศึกษาถึงวิธีการเรียกใช้งาน Application Framework ในส่วนที่ต้องการใช้งาน แล้วนำมาใช้งานซึ่งมีหลายกลุ่มด้วยกัน ตัวอย่างเช่น

(2.1) Activities Manager เป็นกลุ่มของชุดคำสั่งที่จัดการเกี่ยวกับวงจรการทำงานของหน้าต่างโปรแกรม (Activity)

(2.2) Content Providers เป็นกลุ่มของชุดคำสั่ง ที่ใช้ในการเข้าถึงข้อมูลของโปรแกรมอื่น และสามารถแบ่งปันข้อมูลให้โปรแกรมอื่นเข้าถึงได้

(2.3) View System เป็นกลุ่มของชุดคำสั่งที่เกี่ยวกับการจัดการโครงสร้างของหน้าจอที่แสดงผลในส่วนที่ติดต่อกับผู้ใช้งาน (User Interface)

(2.4) Telephony Manager เป็นกลุ่มของชุดคำสั่งที่ใช้ในการเข้าถึงข้อมูลด้านโทรศัพท์ เช่น หมายเลขโทรศัพท์ เป็นต้น

(2.5) Resource Manager เป็นกลุ่มของชุดคำสั่งในการเข้าถึงข้อมูลที่เป็น ข้อความหรือรูปภาพ

(2.6) Location Manager เป็นกลุ่มของชุดคำสั่งที่เกี่ยวกับตำแหน่งทางภูมิศาสตร์ที่ระบบปฏิบัติการได้รับค่าจากอุปกรณ์

(2.7) Notification Manager เป็นกลุ่มของชุดคำสั่งที่จะถูกเรียกใช้เมื่อโปรแกรมต้องการแสดงผลให้กับผู้ใช้งาน ผ่านทางแถบ (Status Bar) ของหน้าจอ

(3) Libraries

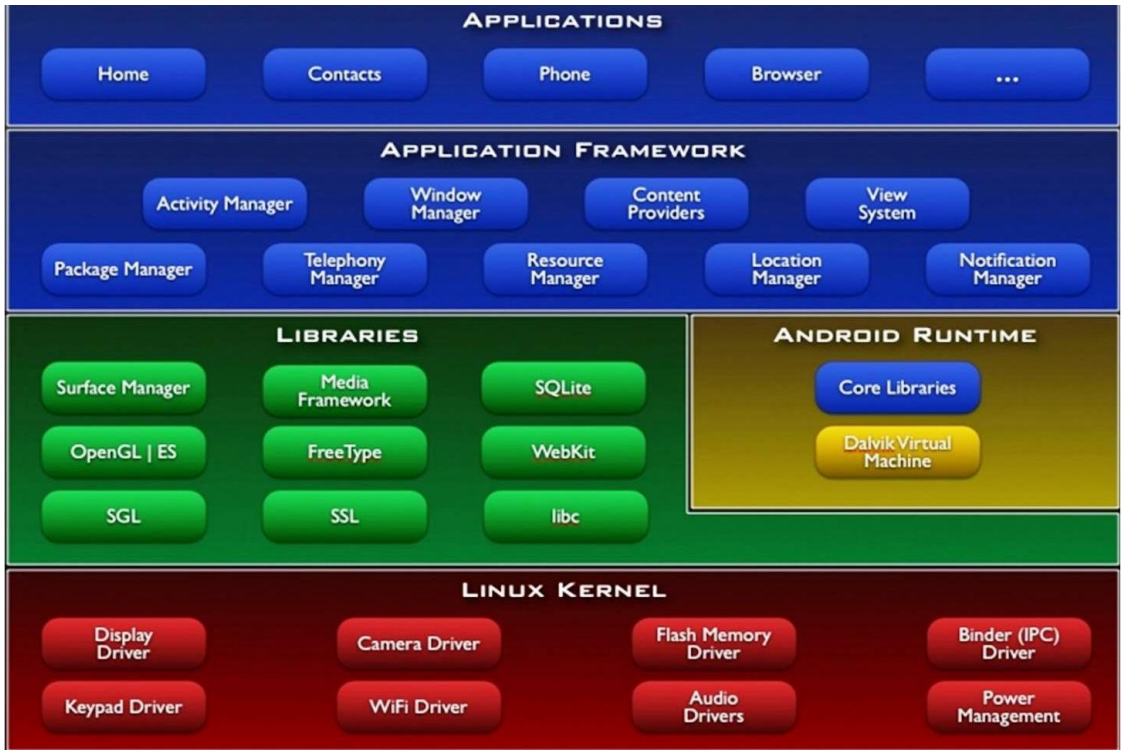
เป็นส่วนของชุดคำสั่งที่พัฒนาด้วย C/C++ โดยแบ่งชุดคำสั่งออกเป็นกลุ่มตามวัตถุประสงค์ของการใช้งาน เช่น Surface Manage จัดการเกี่ยวกับการแสดงผล Media Framework จัดการเกี่ยวกับการแสดงภาพและเสียง Open GL I ES และ SGL จัดการเกี่ยวกับภาพ 3 มิติ และ 2 มิติ SQLite จัดการเกี่ยวกับระบบฐานข้อมูล เป็นต้น

(4) Android Runtime

จะมี Darvik Machine ที่ถูกออกแบบมา เพื่อให้ทำงานบนอุปกรณ์ที่มีหน่วยความจำ (Memory) หน่วยประมวลผลกลาง (CPU) และพลังงาน (Battery) ที่จำกัด ซึ่งการทำงานของ Darvik Machine จะทำการแปลงไฟล์ที่ต้องการทำงาน ไปเป็นไฟล์ .DEX ก่อนการทำงาน เหตุผลก็เพื่อให้มีประสิทธิภาพเพิ่มขึ้น เมื่อใช้งานกับหน่วยประมวลผลกลางที่มีความเร็วไม่มาก ส่วนต่อมาก็คือ Core Libraries ที่เป็นส่วนรวบรวมคำสั่งและชุดคำสั่งสำคัญ โดยถูกเขียนด้วยภาษาจาวา (Java Language)

(5) Linux Kernel

เป็นส่วนที่ทำหน้าที่เป็นหัวใจสำคัญ ในการจัดการกับบริการหลักของระบบปฏิบัติการ เช่น เรื่องหน่วยความจำ พลังงาน ติดต่อกับอุปกรณ์ต่างๆ ความปลอดภัย เครือข่าย โดยแอนดรอยด์ได้นำเอาส่วนนี้มาจากระบบปฏิบัติการลินุกซ์ รุ่น 2.6 (Linux 2.6 Kernel) ซึ่งได้มีการออกแบบมาเป็นอย่างดี



รูปที่ 2.36 โครงสร้างต่าง ๆ ที่เป็นองค์ประกอบสำคัญในระบบปฏิบัติการแอนดรอยด์

ที่มา: รุ่งโรจน์ เกื้อกูลพงศ์. วิวัฒนาการของระบบปฏิบัติการแอนดรอยด์ (Android) สู่เทคโนโลยีในอนาคต. เซมิคอนดักเตอร์ อิเล็กทรอนิกส์. 2556:136.

2.1.9 App Inventor

โปรแกรม MIT APP Inventor เป็นโปรแกรมสำหรับสร้างแอพพลิเคชั่นบนโทรศัพท์มือถือแอนดรอยด์และแทบเลต ที่อำนวยความสะดวกให้กับนักพัฒนาในรูปแบบของการประกอบแบบลอจิคัลบล็อก (Logical Block) ในลักษณะเดียวกับโปรแกรมภาษา Scratch Visual Programming Language ซึ่งเป็นการนำวิชั่วบล็อก (Visual Blocks) มาประกอบเป็นลอจิกคำสั่งแทนการพัฒนาซอฟต์แวร์ ทำให้ผู้ที่สนใจจะพัฒนาซอฟต์แวร์บนอุปกรณ์โทรศัพท์แอนดรอยด์ไม่จำเป็นต้องมีความรู้เรื่องการโปรแกรมภาษามาก่อนก็สามารถเริ่มต้นการพัฒนาซอฟต์แวร์สำหรับโทรศัพท์แอนดรอยด์ได้โดยง่าย

รูปที่ 2.37 ลักษณะการใช้งาน MIT App Inventor

ที่มา: http://appinventor.mit.edu/explore/ai2/setup.html

2.2 งานวิจัยที่เกี่ยวข้อง

ณัฐธิดา มากสังข์ และจินดารัตน์ วรรณคุณ (2556) ได้ทำการศึกษาระบบตรวจสอบอุณหภูมิด้วยสมาร์ทโฟนผ่านระบบปฏิบัติการแอนดรอยด์ เป็นโครงงานที่เกี่ยวกับแอพพลิเคชั่นตรวจสอบอุณหภูมิ โดยมีการส่งข้อมูลผ่านการเชื่อมต่อแบบไร้สาย ซึ่งประกอบไปด้วยตัวตรวจรู้สำหรับวัดอุณหภูมิ และส่งค่าของอุณหภูมิมาทำการประมวลผลในไมโครคอนโทรลเลอร์ ATMEGA1280 หลังจากนั้นจะส่งข้อมูลมาให้กับบอร์ดอาร์ดูโน่ ไวไฟ ชิว เพื่อทำการส่งข้อมูลที่ได้รับมาด้วยการสื่อสารแบบไร้สาย หลังจากนั้นแอพพลิเคชั่นที่เขียนโดยโปรแกรมอีคิปส์ จะทำการประมวลผลค่าที่ได้รับมา แล้วนำมาแสดงผลทางหน้าจอสมาร์ทโฟน โดยค่าของอุณหภูมิจะอยู่ในหน่วยองศาเซลเซียส อีกทั้งยังสามารถตั้งค่าขอบเขตของอุณหภูมิเพื่อให้มีการเตือนสำหรับงานที่ต้องการควบคุมอุณหภูมิ จากการออกแบบแอพพลิเคชั่นด้วยภาษาจาวา จะแบ่งช่วงของอุณหภูมิได้ทั้งหมด 3 แบบ คือ อุณหภูมิต่ำกว่าที่ตั้งไว้ สูงกว่าที่ตั้งไว้ และอุณหภูมิปกติ คือ อยู่ในขอบเขตที่เรากำหนดไว้ โดยแอพพลิเคชั่นจะมีการทำงานโดยแจ้งเตือนด้วยข้อความปรากฏที่หน้าจอสมาร์ทโฟน พร้อมทั้งมีการแจ้งเตือนด้วยเสียง หากอุณหภูมิที่กำลังวัดอยู่นั้นไม่ได้อยู่ในขอบเขตที่กำหนดไว้ โดยค่าที่ได้จากเครื่องวัดอุณหภูมิจะมีค่าใกล้เคียงกับค่ามาตรฐานเมื่อเทียบกับเทอร์โมมิเตอร์แบบดิจิตอลรุ่น WT-1

กรรภิรมย์ สุวรรณประสิทธิ์ และรุ่งฤดี ธรรมขันธ์ (2556) ได้ทำการศึกษาการควบคุมอุปกรณ์ไฟฟ้าผ่าแอพพลิเคชั่นบนระบบแอนดรอยด์ เป็นการสร้างแอพพลิเคชั่นควบคุม ปิด เปิดหลอดไฟส่องสว่างภายในบ้าน โดยสั่งงานบนระบบปฏิบัติการแอนดรอยด์ผ่านเครือข่ายอินทราเน็ตไร้สาย ไปยังบอร์ด Embedded system โดยใช้ระบบปฏิบัติการณ์ Linux ควบคุมการเปิด ปิดหลอดไฟส่องสว่าง ซึ่งได้ออกแบบตัวโปรแกรมให้สามารถควบคุมอุปกรณ์ไฟฟ้าจากระยะไกลผ่านเครือข่ายวายฟาย ทำให้ผู้ใช้สามารถควบคุมการเปิด ปิด หลอดไฟส่องสว่างผ่านแอพพลิเคชั่นได้ โดยในส่วนของแอพพลิเคชั่น สามารถสั่งงานผ่านทางโทรศัพท์มือถือบนระบบปฏิบัติการแอนดรอยด์ผ่านเครือข่ายไร้สายวายฟาย โดยระบุ IP Address ของปลายทาง จากนั้นจึงส่งคำสั่ง On เพื่อเปิดหลอดส่องสว่าง ส่งคำสั่ง Off เพื่อปิดหลอดส่องสว่าง ซึ่งแต่ละครั้งที่มีการสั่งงาน หน้าแอพพลิเคชั่นจะแสดงสถานะของอุปกรณ์ปลายทาง เช่น เมื่อสั่งเปิดไฟหลอดที่ 1 หน้าแอพพลิเคชั่นจะแสดงข้อความ LED1 On เมื่อสั่งปิดไฟ หน้าแอพพลิเคชั่นจะแสดงข้อความ LED1 Off สามารถเปิด ปิด อุปกรณ์ได้สะดวกสบายและประหยัดเวลาในการสั่งงานเครื่องใช้ไฟฟ้า

ธีรภัทร ผิวทน ภัทร บุญวงษ์ และวัชรชัย วิริยะสุทธิวงศ์ (2557) ได้ทำการศึกษาการตรวจวัดและแสดงอุณหภูมิของอุปกรณ์ไฟฟ้า ซึ่งเป็นอีกช่องทางหนึ่งที่สามารถช่วยป้องกันการเกิดเพลิงไหม้ได้ โครงงานนี้ได้มีการนำเอาระบบสวิทช์เปิดปิดและการตรวจวัดอุณหภูมิของอุปกรณ์ไฟฟ้ามาประยุกต์ใช้บนเครือข่ายอินเตอร์เน็ต โดยแสดงอุณหภูมิของอุปกรณ์ไฟฟ้าผ่านอินเตอร์เน็ตและแจ้งเตือนความผิดปกติของอุณภูมิของอุปกรณ์ไฟฟ้าผ่านอีเมลล์ เพื่อเป็นการป้องกันการเกิดเพลิงไหม้จากไฟฟ้าลัดวงจร โดยใน

ส่วนของการทดลองได้แบ่งการทดลองออกเป็นสองส่วน คือ ส่วนของการปิดเปิดสวิทซ์ไฟฟ้าผ่านอินเตอร์เน็ต โดยทำการปิดเปิดอุปกรณ์ไฟฟ้าทั้งหมด 4 ชนิด คือ โคมไฟ พัดลม กระติกน้ำร้อน และเตารีดไฟฟ้า และส่วนที่สองคือการทดลองการวัดอุณหภูมิ ทำการทดลองโดยนำตัวตรวจรู้ DS18B20 ไปติดตั้งกับอุปกรณ์ไฟฟ้า และยังติดตั้งเทอร์โมมิเตอร์บนอุปกรณ์ไฟฟ้าด้วย เพื่อวัดอุณหภูมิให้เป็นค่าเปรียบเทียบกับ DS18B20 โดยได้ทำการทดลองกับอุปกรณ์ไฟฟ้าทั้งหมด 3 ชนิด คือ กระติกน้ำร้อนไฟฟ้า เตารีดไฟฟ้า และกระทะไฟฟ้า โดยค่าที่วัดได้โปรแกรมสามารถนำไปแสดงผลได้ พร้อมทั้งยังสามารถนำค่าอุณหภูมิไปเปรียบเทียบกับค่าอุณหภูมิที่ได้กำหนดไว้และนำไปแจ้งเตือนผ่านอีเมลล์ได้

จากการศึกษาข้อมูลทั้งหมด เราได้พบว่าการทำการทดลองในส่วนของอุปกรณ์ฮาร์ดแวร์ ที่ใช้บอร์ดไมโครคอนโทรลเลอร์ที่มีข้อจำกัดในภาษาที่ใช้ในการเขียนโปรแกรม ยกตัวอย่างเช่น บอร์ดราสเบอร์รี่ไพ ที่ต้องใช้ภาษา Phyton ในการเขียนโปรแกรม เมื่อถูกนำมาประยุกต์ใช้งานร่วมกับอุปกรณ์ในส่วนของฮาร์ดแวร์ที่ใช้ภาษาที่แตกต่างกันออกไปในการเขียนโปรแกรมจะทำให้เกิดความยุ่งยากในการทำการทดลอง และทำให้เกิดความล่าช้าของการส่งข้อมูลระหว่างอุปกรณ์ และการทำการส่งข้อมูลผ่านเครือข่ายแบบใช้สาย หรือ LAN นั้นมีความล่าช้าในการส่งข้อมูลเนื่องจากมีอัตราการส่งข้อมูลช้าและมีความยุ่งยากหากเกิดการเคลื่อนย้ายอุปกรณ์ และนอกจากนั้นหากทำการทดลองผ่านระบบเครือข่ายไร้สายแบบไวเลสก็จะพบข้อเสียคือข้อจำกัดในเรื่องของระยะทางในการติดต่อสื่อสาร

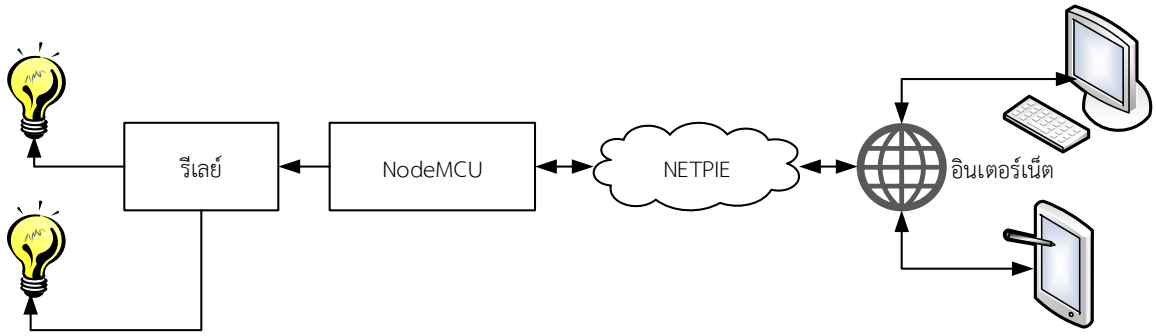
ดังนั้นในโครงงานเล่มนี้จึงได้เลือกใช้อุปกรณ์ฮาร์ดแวร์ เป็น NodeMCU ที่เป็นอุปกรณ์แบบโอเพ่นซอร์ส ไม่มีความยุ่งยากในการเขียนโปรแกรมมาใช้งานร่วมกับรีเลย์และตัวตรวจรู้ PZEM-004T และทำงานโดยใช้การเชื่อมต่อผ่านระบบประมวลผลกลุ่มเมฆ ทำให้เมื่อใดก็ตามที่อุปกรณ์มีการเชื่อมต่ออินเตอร์เน็ตจะสามารทำการควบคุมปิดเปิดอุปกรณ์ไฟฟ้า และยังสามารถดูข้อมูลการใช้พลังงานได้ผ่านทางเว็บแอพพลิเคชั่น

บทที่ 3
วิธีการดำเนินงาน

ในบทนี้จะกล่าวถึงการสร้างตัวควบคุมอุปกรณ์ไฟฟ้าและบันทึกข้อมูลการใช้พลังงาน โดยใช้งานผ่านระบบอินเตอร์เน็ตของทุกสรรพสิ่ง (Internet of Things: IoT) โดยแยกการออกแบบและการสร้างออกเป็นส่วนหลัก ๆ ได้ดังนี้

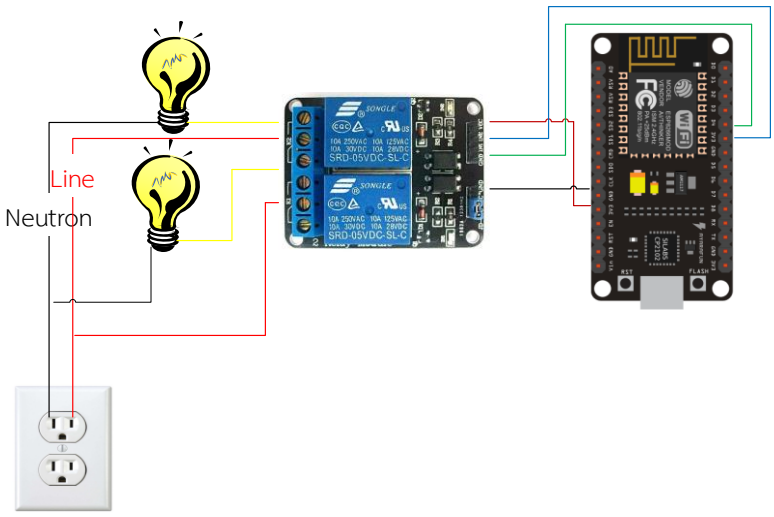
3.1 ตัวควบคุมอุปกรณ์ไฟฟ้า

3.1.1 บล็อกไดอะแกรมตัวควบคุมอุปกรณ์ไฟฟ้า



รูปที่ 3.1 บล็อกไดอะแกรมควบคุมอุปกรณ์ไฟฟ้า

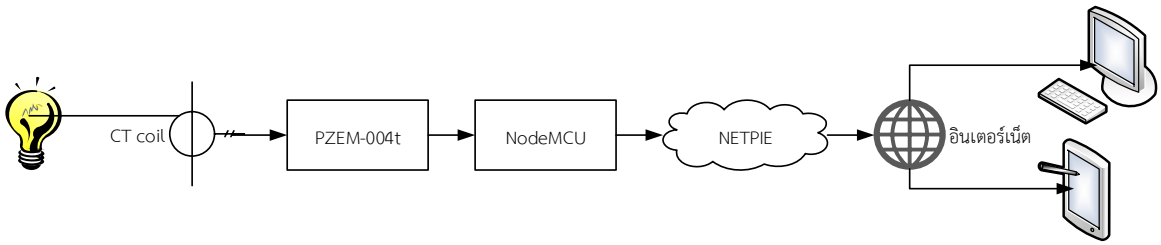
3.1.2 ฮาร์ดแวร์



รูปที่ 3.2 การเชื่อมต่ออุปกรณ์ในส่วนของการควบคุมอุปกรณ์ไฟฟ้า

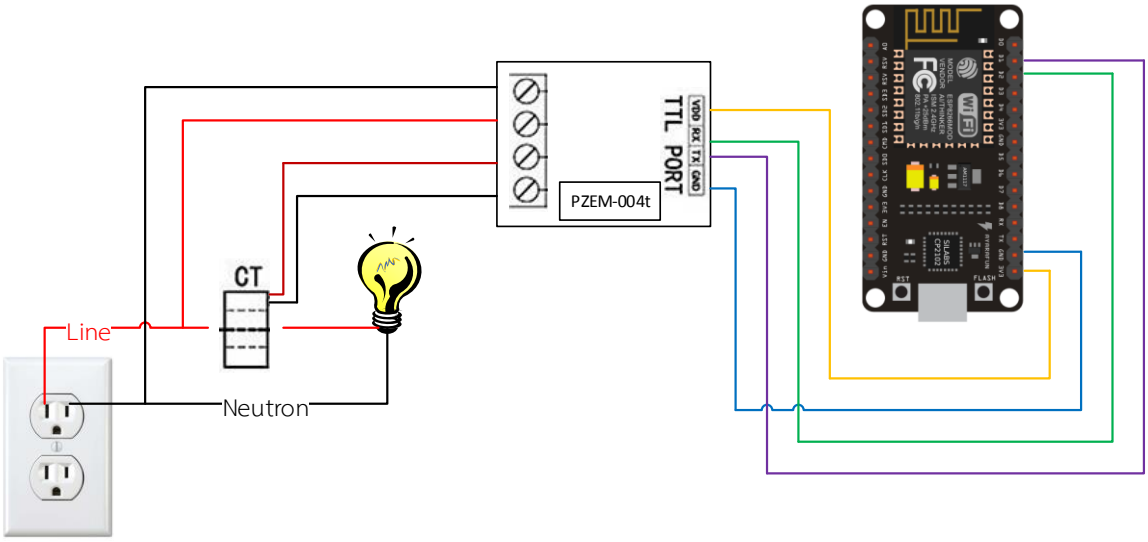
3.2 ตัวบันทึกข้อมูลการใช้พลังงาน

3.2.1 บล็อกไดอะแกรมตัวบันทึกข้อมูลการใช้พลังงาน



รูปที่ 3.3 บล็อกไดอะแกรมตัวบันทึกข้อมูลการใช้พลังงาน

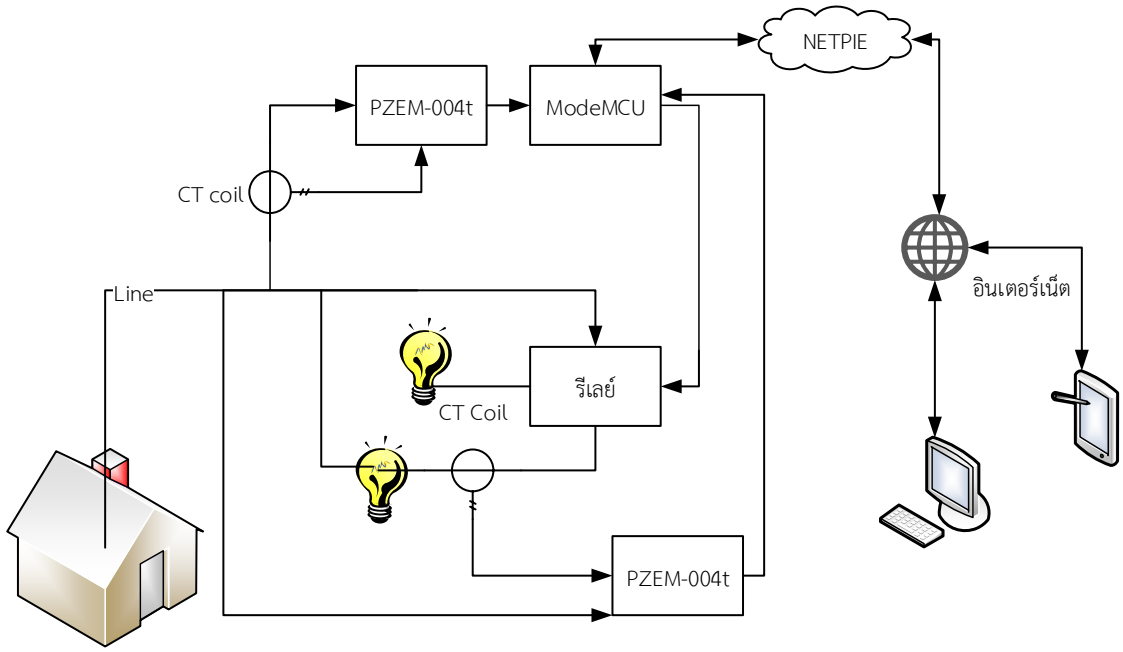
3.2.2 ฮาร์ดแวร์



รูปที่ 3.4 การเชื่อมต่ออุปกรณ์ในส่วนของตัวบันทึกข้อมูลการใช้พลังงาน

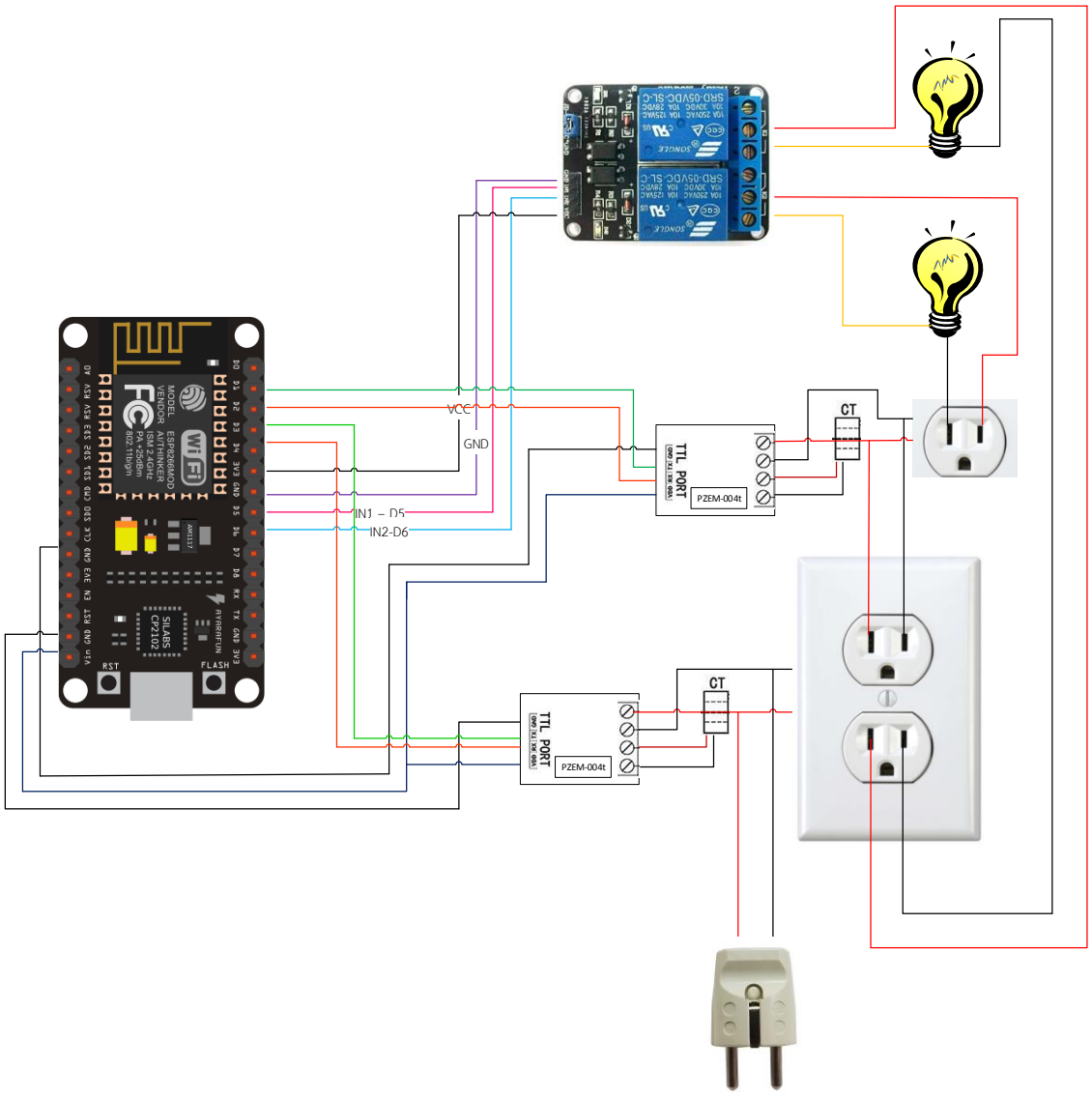
3.3 ตัวควบคุมอุปกรณ์ไฟฟ้าและบันทึกข้อมูลการใช้พลังงาน

3.3.1 ภาพรวมของตัวควบคุมอุปกรณ์ไฟฟ้าและบันทึกข้อมูลการใช้พลังงาน



รูปที่ 3.5 ภาพรวมของตัวควบคุมอุปกรณ์ไฟฟ้าและบันทึกข้อมูลการใช้พลังงาน

3.3.2 ฮาร์ดแวร์ในส่วนของภาคส่ง

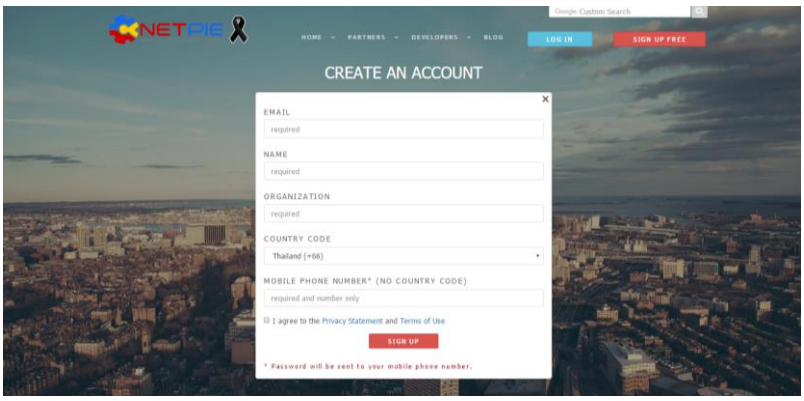


รูปที่ 3.6 การเชื่อมต่ออุปกรณ์ ฮาร์ดแวร์ในส่วนของภาคส่ง

3.3.3 เน็ตพาย (NETPIE)

3.3.3.1 การสมัครใช้งาน

(1) ไปที่เว็บไซต์ http//:netpie.io/sign_up จะปรากฏหน้าเว็บดังรูปที่ 3.7 กรอกข้อมูลให้เรียบร้อยจากนั้นคลิกที่ปุ่ม SIGN UP เพื่อยืนยันการลงทะเบียน



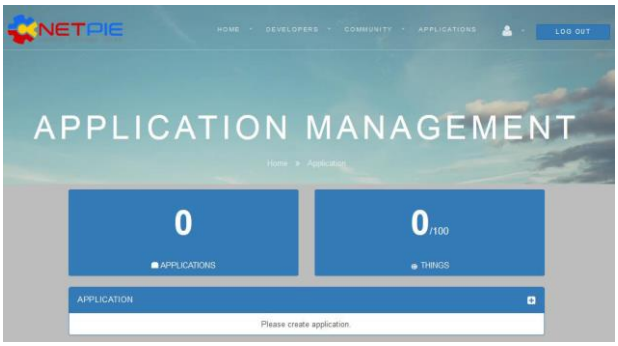
รูปที่ 3.7 หน้าลงทะเบียนผู้ใช้งานเน็ตพาย (NETPIE)

(2) รอรับ SMS จากทางเน็ตพาย (NETPIE) ซึ่งส่งไปยังหมายเลขโทรศัพท์เคลื่อนที่ที่ลงทะเบียนไว้ ตัวอย่าง SMS: Your one-time password for NETPIE is 258771949728

(3) คลิกที่เมนู LOGIN เพื่อเข้าสู่ระบบนำอีเมลล์ที่ลงทะเบียนไว้ใส่ในช่อง USERNAME OR EMAIL ADDRESS และนำรหัสผ่านที่ได้รับจาก SMS ใส่ในช่อง PASSWORD แล้วคลิกปุ่ม LOGIN

(4) ตั้งรหัสผ่านใหม่ โดยใส่รหัสผ่านใหม่ในช่อง NEW PASSWORD และ CONFIRM PASSWORD

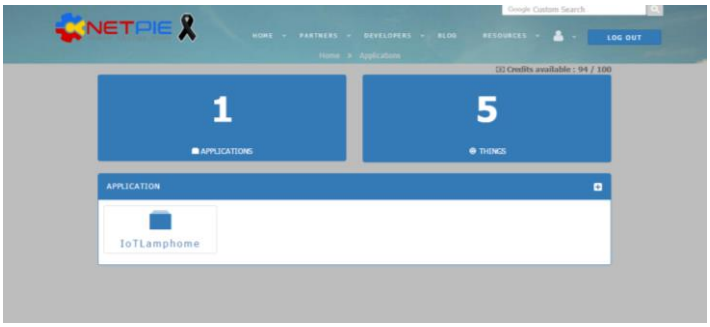
(5) หลังจากสมัครสมาชิก และล็อกอินเรียบร้อยแล้ว สามารถสร้างแอพพลิเคชั่นได้โดยเลือกที่เมนู APPLICATIONS เพื่อเข้าไปที่หน้า APPLICATION MANAGEMENT ซึ่งแสดงแอพพลิเคชั่นทั้งหมดที่ผู้ใช้งานมีอยู่ การสร้างแอพพลิเคชั่นใหม่ทำได้โดยการคลิกที่ปุ่ม +



รูปที่ 3.8 หน้าจัดการแอพพลิเคชั่นสำหรับผู้ใช้งานเน็ตพาย (NETPIE)

(6) ใน Popup ให้กำหนดชื่อแอพพลิเคชั่น (Application ID หรือ APPID) ที่ต้องการ ในที่นี้ใช้ชื่อว่า IoTLamphome จากนั้นคลิกปุ่ม CREAT การตั้งชื่อ AppID จะต้องไม่ซ้ำกับผู้อื่น ดังนั้น ควรเลือกชื่อที่มีความเฉพาะตัวเพื่อให้จำได้ เนื่องจาก AppID นี้จะถูกพัฒนาโปรแกรมต่อไป

(7) หากสร้างแอพพลิเคชั่นใหม่ได้สำเร็จ ที่หน้าต่าง APPLICTION ในหน้า APPLICATION MANAGEMENT จะปรากฏช่องของชื่อ AppID ที่เรากำหนดไว้ หากชื่อซ้ำหรือมีข้อผิดพลาดจะมีข้อความแจ้งเตือนขึ้นมาว่า This AppID is already in use. Please select another AppID หลังจากสร้างเสร็จ จะเปลี่ยนไปที่หน้าของแอพพลิเคชั่นที่สร้างขึ้นใหม่นี้โดยอัตโนมัติ

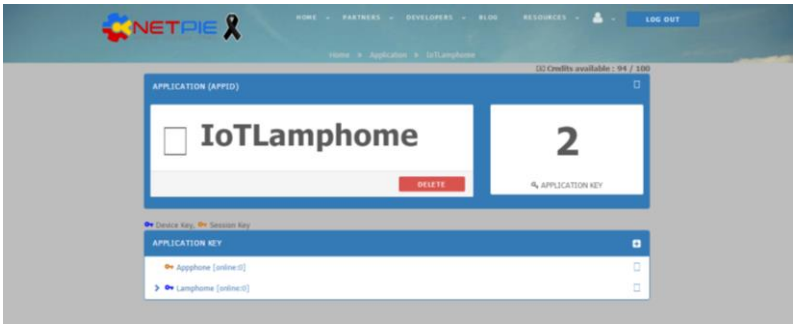


รูปที่ 3.9 ตัวอย่างหน้าแอพพลิเคชั่นหนึ่งของเน็ตพาย (NETPIE)

(8) สร้าง Application Key โดนคลิกที่ปุ่ม +

(9) กำหนดชื่อของ Application Key ตามต้องการ ในที่นี้ใช้ข้อความว่า Lamphome และเลือกชนิดของ Key ใน Drop-down box ให้เป็น Device Key เมื่อสร้างเสร็จ จะมีชื่อ Lamphome ปรากฏในหน้าต่าง APPLICATION KEY

(10) กำหนดชื่อของ Application Key ตามต้องการ ในที่นี้ใช้ข้อความว่า Appphone และเลือกชนิดของ Key ใน Drop-down box ให้เป็น Session Key เมื่อสร้างเสร็จ จะมีชื่อ Appphone ปรากฏในหน้าต่าง APPLICATION KEY



รูปที่ 3.10 การตั้งชื่อและเลือกชนิด Application Key

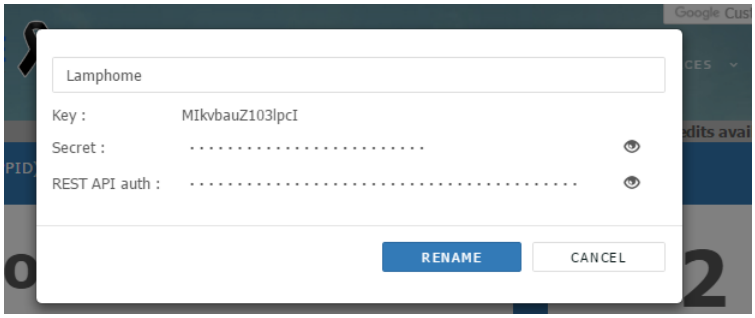
(11) ตรวจสอบข้อมูลหรือดูละเอียดคีย์ด้วยการคลิกที่คีย์แต่ละตัวจะปรากฏข้อมูลของ Application Key ซึ่งประกอบด้วย

(11.1) Application Key Name สามารถเปลี่ยนแปลงชื่อได้ตามที่ต้องการ ซึ่งชื่อนี้จะตั้งซ้ำกับชื่อผู้อื่นได้ และไม่ได้นำไปใช้แทนชื่อ AppID

(11.2) Key เป็นกุญแจที่ใช้สำหรับอุปกรณ์เชื่อมต่อเน็ตพาย (NETPIE)

(11.3) Secret เป็นรหัสลับหรือรหัสผ่านที่ใช้คู่กับคีย์สำหรับอุปกรณ์เชื่อมต่อเน็ตพาย (NETPIE)

(11.4) REST API auth เป็นกุญแจที่เรียงต่อกับรหัสลับเพื่อใช้สำหรับการใช้งานด้วย REST API



รูปที่ 3.11 รายละเอียดของคีย์

3.3.3.2 การสร้างฟรีบอร์ดของเน็ตพาย (NETPIE Freeboard)

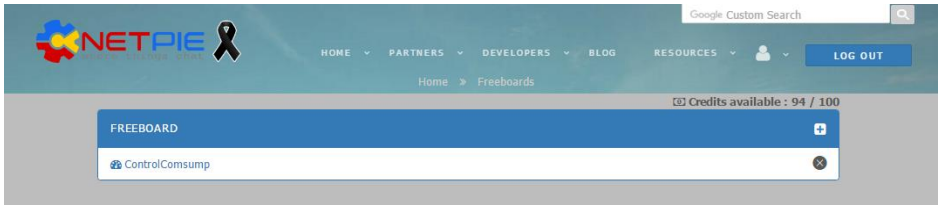
Freeboard เป็น Web Application ที่สามารถสร้างแดชบอร์ด (Dashboard) เพื่อแสดงผลสำหรับแอพพลิเคชั่นอินเตอร์เน็ตของทุกสรรพสิ่ง โดยสามารถใช้เป็นกระดานส่วนตัว สามารถวางปุ่มกดสวิตช์ไว้ใช้สำหรับควบคุมอุปกรณ์ หรือ วางหน้าปัดเพื่อแสดงผลข้อมูลต่างๆ ที่ได้จากอุปกรณ์ เช่น ตัวตรวจรู้ในระบบอินเตอร์เน็ตของทุกสรรพสิ่ง นอกจากนี้ยังสามารถแสดงผลเป็นกราฟได้ ส่วนหน้ากระดานหรือแดชบอร์ดนั้น สามารถปรับแต่งได้โดยง่าย เพียงแค่ป้อนข้อมูลเข้าหรือกำหนดคำสั่งก็สามารถทำงานได้แล้ว โดยที่ผู้ใช้ไม่จำเป็นต้องเรียน HTML Web Page เอง และที่สำคัญคือข้อมูลนั้นมีการอัพเดทแบบเวลาจริง (Real-time) มีความเสถียรและเชื่อถือได้ และเป็นโอเพ่นซอร์ส (Open-Source) ซึ่งทำให้นักพัฒนาสามารถต่อยอดให้ดียิ่งขึ้นได้อีกด้วย

NETPIE Freeboard คือฟรีบอร์ดสำหรับการควบคุมและการแสดงผล (Visualization) ข้อมูลที่ดึงมาจากอุปกรณ์ที่ต่อกับเน็ตพาย ทีมงานได้พัฒนา Widget Plugins ขึ้นมาให้ผู้ใช้งานสามารถทำตามความต้องการได้หลากหลาย เช่น สามารถสร้างปุ่มควบคุมและใส่คำสั่ง Javascript สำหรับ Action

ต่างๆ ได้ การใช้งาน เน็ตพายฟรีบอร์ดนั้นสามารถใช้เบราเซอร์เปิดไฟล์ index.html ที่ได้จากการติดตั้งเน็ตพายฟรีบอร์ดหรือสามารถใช้ฟรีบอร์ดทางหน้าเว็บของเน็ตพาย

รูปที่ 3.12 เริ่มต้นการสร้างฟรีบอร์ดบนเน็ตพาย

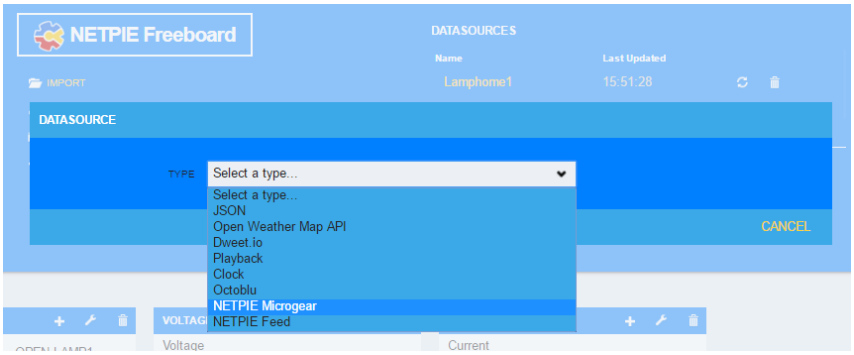
คลิกเครื่องหมาย + เพื่อสร้าง Freeboard ขึ้นมาใหม่ ตั้งชื่อ Freeboard ในที่นี่ตั้งชื่อว่า ControlConsump แล้วกดปุ่ม CREAT



รูปที่ 3.13 ตั้งชื่อ Freeboard ว่า ControlConsump

(1) ในส่วนของการควบคุม เปิด ปิด หลอดไฟดวงที่ 1

(1.1) ด้านล่าง DATASOURCE คลิกที่ ADD จะปรากฏ Datasource Type ชนิดต่าง ๆ ให้เลือกเป็น NETPIE Microgear



รูปที่ 3.14 เลือกประเภทของ Datasource เป็น NETPIE Microgear

(1.2) ใส่ข้อมูลสำหรับ Datasource ซึ่งประกอบด้วย

(1.2.1) NAME คือ ชื่อเรียก datasource ที่ใช้อ้างอิง ไม่เกิน 16 ตัวอักษร ในที่นี้ตั้งชื่อว่า Lamphome1

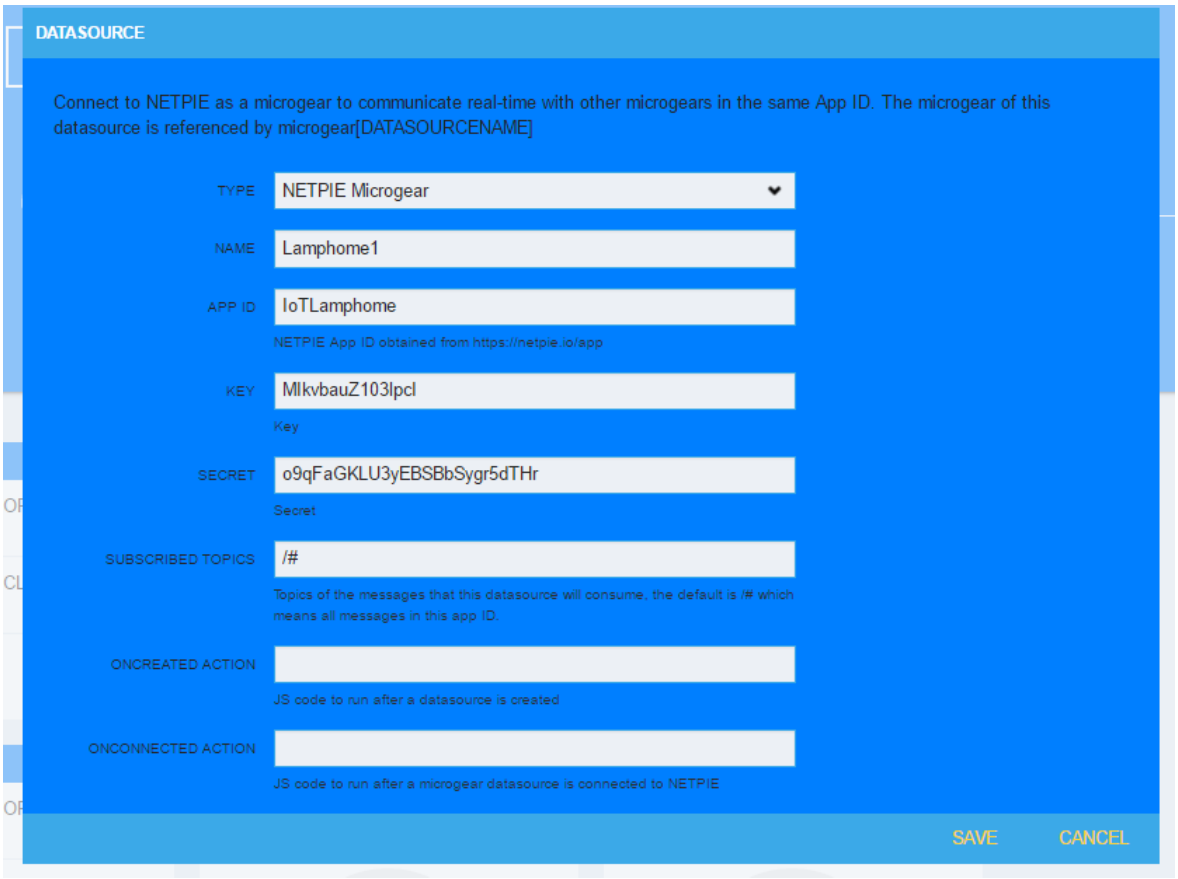
(1.2.2) APP ID คือ App ID ที่ได้สร้างผ่านหน้าเว็บ https://netpie.io/app

(1.2.3) KEY คือ Key ที่ได้จากการสร้าง App Key บนเว็บ NETPIE

(1.2.4) SECRET คือ Secret ของ Key บนเว็บ NETPIE

(1.2.5) SUBSCRIBED TOPIC คือ topic ที่ใช้สำหรับการรับส่งข้อมูลที่อยู่ภายใน APPID นั้น ๆ กรณีนี้ใช้เป็น /# มีความหมายว่า รับข้อความจากทุก Topic

(1.2.6) เมื่อกรอกเสร็จแล้วกด SAVE



รูปที่ 3.15 การระบุข้อมูลของ Datasource สำหรับใช้ในการควบคุม เปิด ปิด หลอดไฟดวงที่ 1

(1.3) เพิ่ม Panel สำหรับสร้าง Widget ด้วยการคลิก ADD PANE จะปรากฏ Panel เพิ่มขึ้นมาด้านล่าง

(1.4) เพิ่ม Widget บน Panel ที่สร้างขึ้นใหม่โดย คลิกที่เครื่องหมาย + และเลือกชนิดของ Widget โดยเลือกเป็น Button

(1.5) ใส่ข้อมูลสำหรับ Widget ซึ่งประกอบด้วย

(1.5.1) TYPE คือ ชนิดของ Widget

(1.5.2) BUTTON CAPTION คือ คำที่จะปรากฏบนปุ่ม

(1.5.3) LABEL TEXT คือ ข้อความที่ปรากฏข้างปุ่มกด

(1.5.4) BUTTON COLOR คือ สีของปุ่มกด

(1.5.5) ONCLICK ACTION

(1.5.6) เมื่อกรอกเสร็จแล้วกด SAVE

รูปที่ 3.16 การระบุข้อมูลของ Widget สำหรับสร้างปุ่มกดเปิดหลอดไฟดวงที่ 1

รูปที่ 3.17 การระบุข้อมูลของ Widget สำหรับสร้างปุ่มกดปิดหลอดไฟดวงที่ 1

(2) ในส่วนของการควบคุม เปิด ปิด หลอดไฟดวงที่ 2

(2.1) ด้านล่าง DATASOURCE คลิกที่ ADD จะปรากฏ Datasource Type ชนิดต่างๆ ให้เลือกเป็น NETPIE Microgear

(2.2) ใส่ข้อมูลสำหรับ Datasource

(2.2.1) NAME คือ ชื่อเรียก datasource ที่ใช้อ้างอิง ไม่เกิน 16 ตัวอักษร ในที่นี้ตั้งชื่อว่า Lamphome2

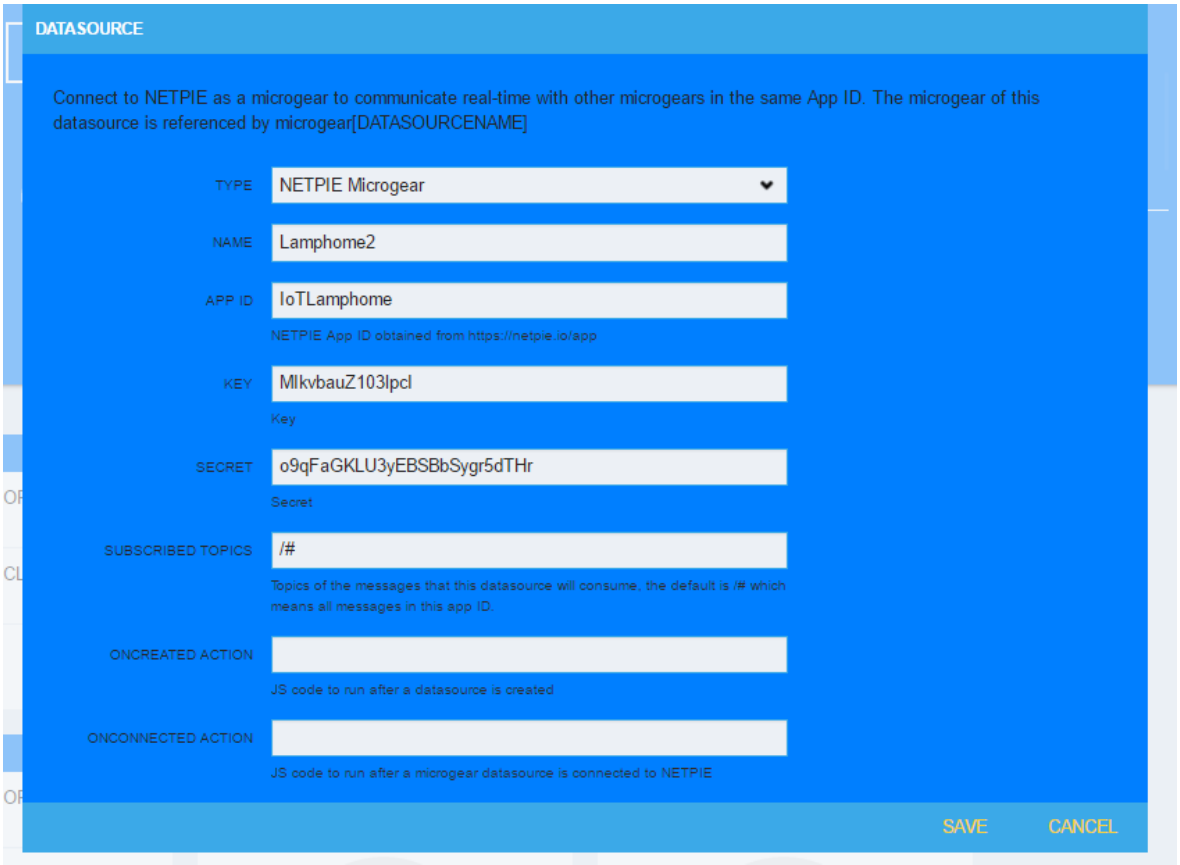
(2.2.2) APP ID คือ App ID ที่ได้สร้างผ่านหน้าเว็บ https://netpie.io/app

(2.2.3) KEY คือ Key ที่ได้จากการสร้าง App Key บนเว็บ NETPIE

(2.2.4) SECRET คือ Secret ของ Key บนเว็บ NETPIE

(2.2.5) SUBSCRIBED TOPIC คือ topic ที่ใช้สำหรับการรับส่งข้อมูลที่อยู่ภายใน APPID นั้น ๆ กรณีนี้ใช้เป็น /# มีความหมายว่า รับข้อความจากทุก Topic

(2.2.6) เมื่อกรอกเสร็จแล้วกด SAVE



รูปที่ 3.18 การระบุข้อมูลของ Datasource สำหรับใช้ในการควบคุม เปิด ปิด หลอดไฟดวงที่ 2

(2.3) เพิ่ม Panel สำหรับสร้าง Widget ด้วยการคลิก ADD PANE จะปรากฏ Panel เพิ่มขึ้นมาด้านล่าง

(2.4) เพิ่ม Widget บน Panel ที่สร้างขึ้นใหม่โดย คลิกที่เครื่องหมาย + และเลือกชนิดของ Widget โดยเลือกเป็น Button

(2.5) ใส่ข้อมูลสำหรับ Widget ซึ่งประกอบด้วย

(2.5.1) TYPE คือ ชนิดของ Widget

(2.5.2) BUTTON CAPTION คือ คำที่จะปรากฏบนปุ่ม

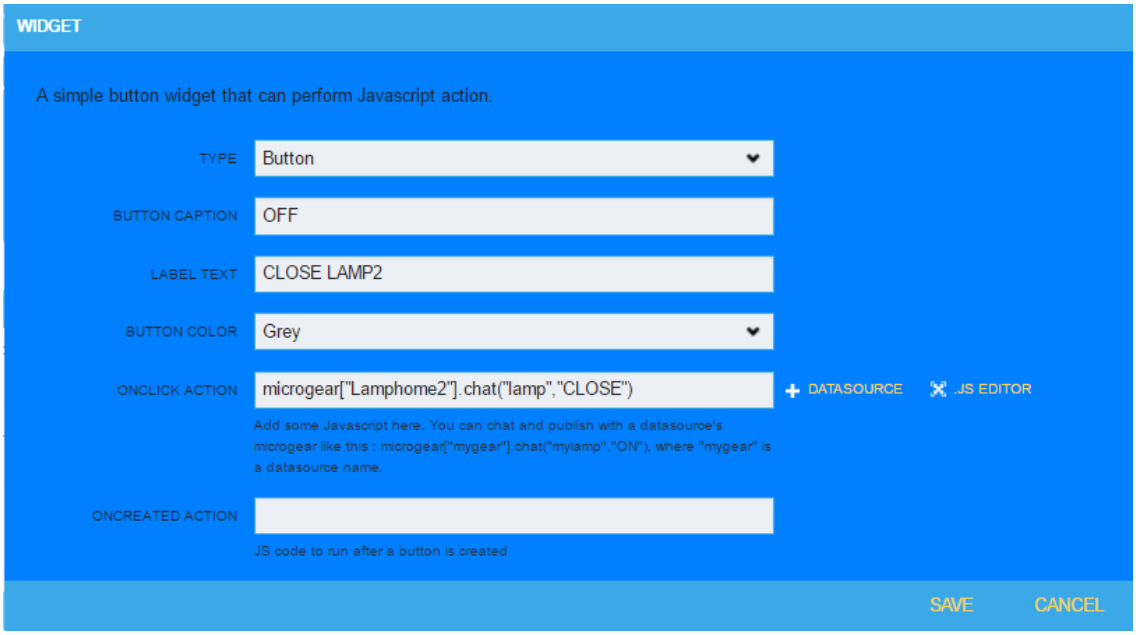
(2.5.3) LABEL TEXT คือ ข้อความที่ปรากฏข้างปุ่มกด

(2.5.4) BUTTON COLOR คือ สีของปุ่มกด

(2.5.5) ONCLICK ACTION

(2.5.6) เมื่อกรอกเสร็จแล้วกด SAVE

รูปที่ 3.19 การระบุข้อมูลของ Widget สำหรับสร้างปุ่มกดเปิดหลอดไฟดวงที่ 2



รูปที่ 3.20 การระบุข้อมูลของ Widget สำหรับสร้างปุ่มกดปิดหลอดไฟดวงที่ 2

(3) ในส่วนของการบันทึกข้อมูลการใช้พลังงานไฟฟ้าแบบแยกส่วน

(3.1) ด้านล่าง DATASOURCE คลิกที่ ADD จะปรากฏ Datasource Type ชนิดต่างๆ ให้เลือกเป็น NETPIE Microgear

(3.2) ใส่ข้อมูลสำหรับ Datasource ซึ่งประกอบด้วย

(3.2.1) NAME คือ ชื่อเรียก datasource ที่ใช้อ้างอิง ไม่เกิน 16 ตัวอักษร ในที่นี่ตั้งชื่อว่า VoltageLamp

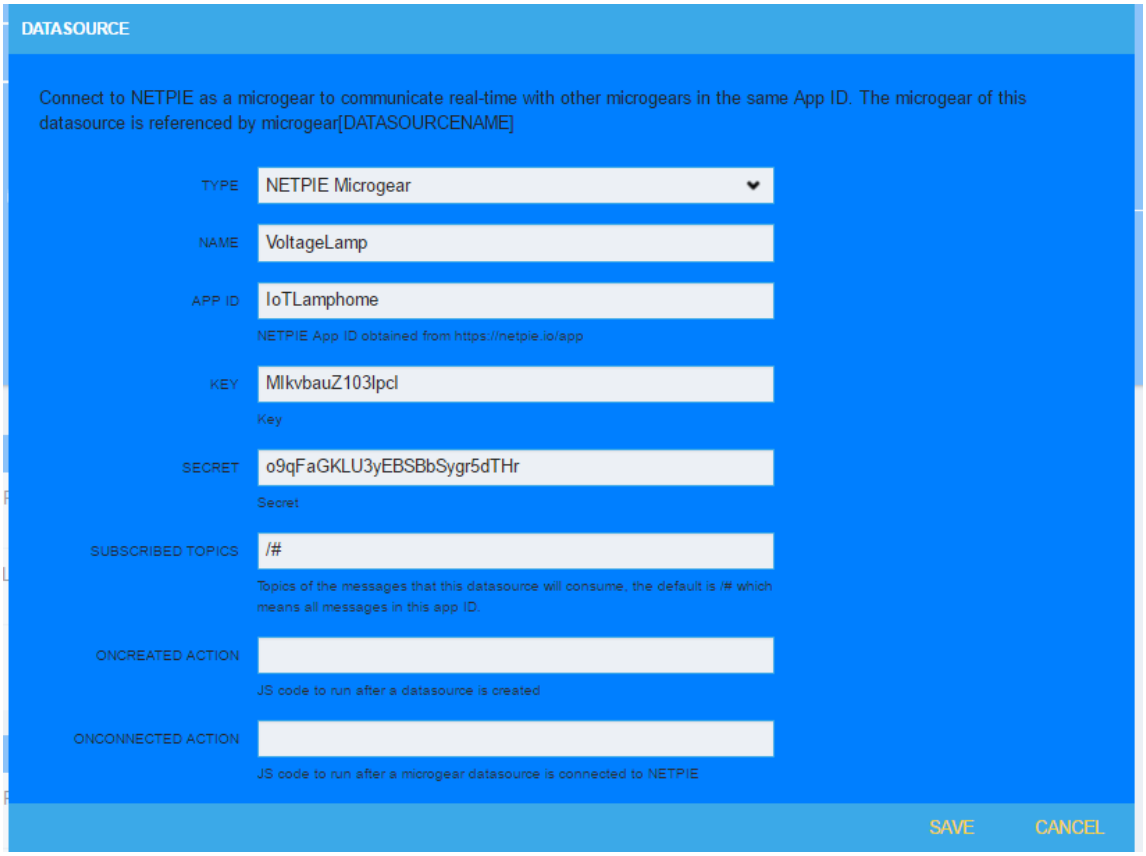
(3.2.2) APP ID คือ App ID ที่ได้สร้างผ่านหน้าเว็บ https://netpie.io/app

(3.2.3) KEY คือ Key ที่ได้จากการสร้าง App Key บนเว็บ NETPIE

(3.2.4) SECRET คือ Secret ของ Key บนเว็บ NETPIE

(3.2.5) SUBSCRIBED TOPIC คือ topic ที่ใช้สำหรับการรับส่งข้อมูลที่อยู่ภายใน APPID นั้น ๆ กรณีนี้ใช้เป็น /# มีความหมายว่า รับข้อความจากทุก Topic

(3.2.6) เมื่อกรอกเสร็จแล้วกด SAVE



รูปที่ 3.21 การระบุข้อมูลของ Datasource สำหรับใช้ในการบันทึกข้อมูลการใช้พลังงานไฟฟ้าแบบแยกส่วน

(3.3) เพิ่ม Panel สำหรับสร้าง Widget ด้วยการคลิก ADD PANE จะปรากฏ Panel เพิ่มขึ้นมาด้านล่าง

(3.4) เพิ่ม Widget บน Panel ที่สร้างขึ้นใหม่โดย คลิกที่เครื่องหมาย + และเลือกชนิดของ Widget โดยเลือกเป็น Gauge

(3.5) ใส่ข้อมูลสำหรับ Widget ซึ่งประกอบด้วย

(3.5.1) TYPE คือ ชนิดของ Widget

(3.5.2) TITLE คือ ชื่อของ Gauge

(3.5.3) VALUE

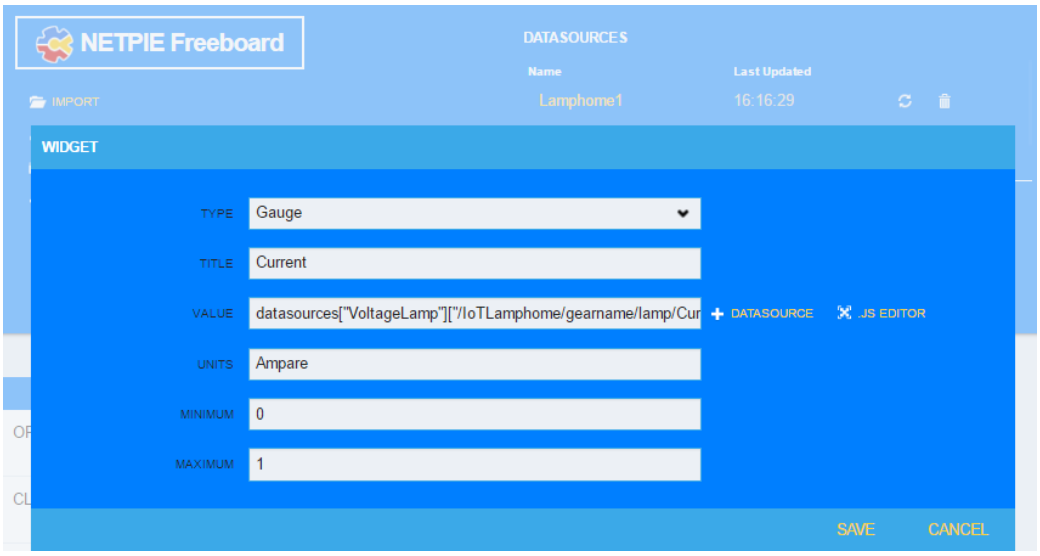
(3.5.4) UNITS คือ หน่วยของข้อมูลที่ทำการบันทึก

(3.5.5) MINIMUM คือค่าเริ่มต้นของข้อมูลหรือค่าต่ำสุดของข้อมูล

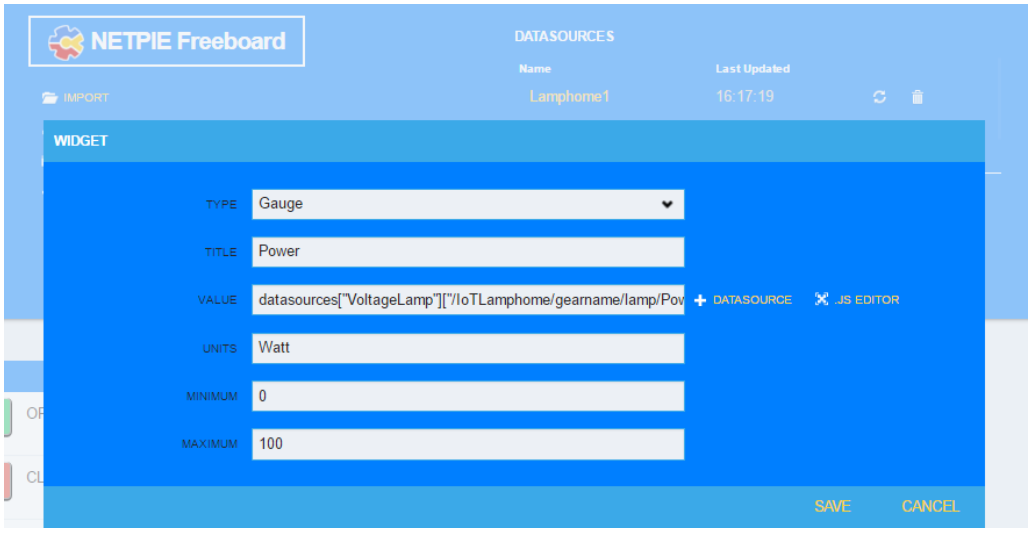
(3.5.6) MAXIMUM คือค่าสูงสุดของข้อมูลที่จะให้แสดงผล

(3.5.7) เมื่อกรอกเสร็จแล้วกด SAVE

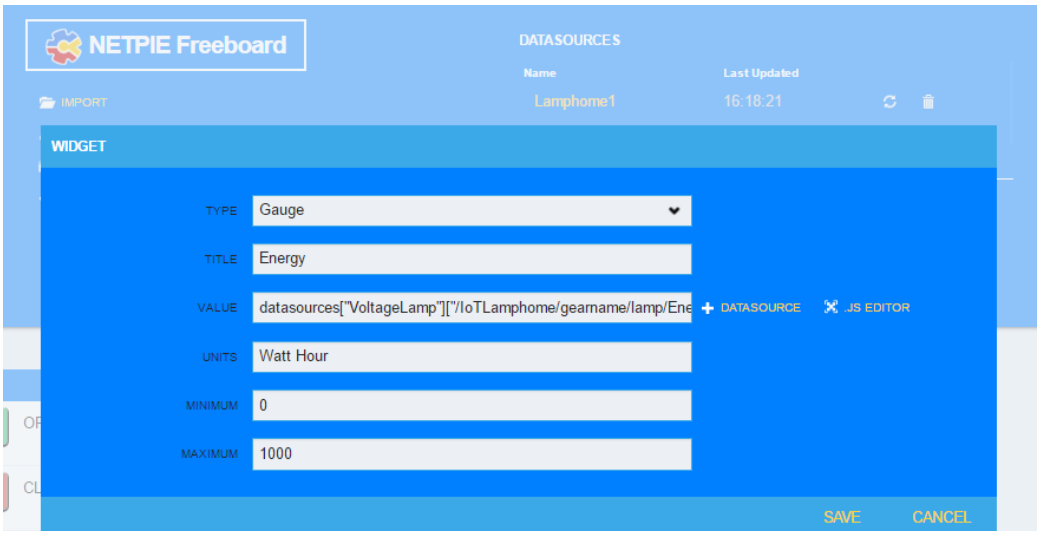
รูปที่ 3.22 การระบุข้อมูลของ Widget สำหรับสร้างการแสดงผลค่าแรงดันไฟฟ้าที่วัดได้แบบแยกส่วน



รูปที่ 3.23 การระบุข้อมูลของ Widget สำหรับสร้างการแสดงผลค่ากระแสไฟฟ้าที่วัดได้แบบแยกส่วน



รูปที่ 3.24 การระบุข้อมูลของ Widget สำหรับสร้างการแสดงผลค่ากำลังไฟฟ้าที่วัดได้แบบแยกส่วน

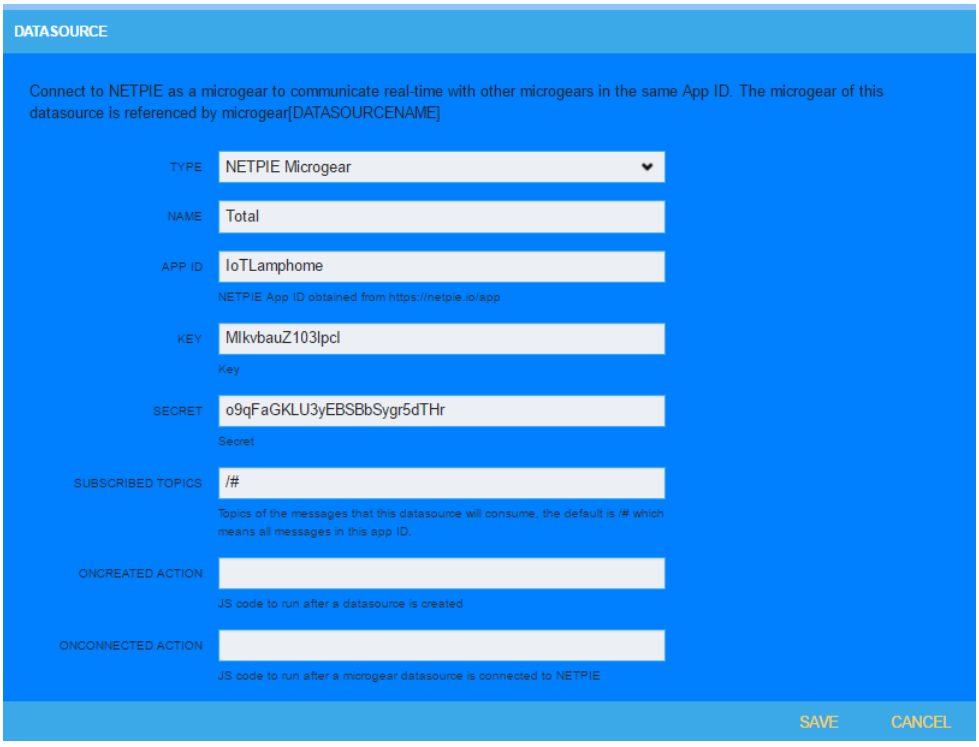


รูปที่ 3.25 การระบุข้อมูลของ Widget สำหรับสร้างการแสดงผลค่าพลังงานไฟฟ้าที่วัดได้แบบแยกส่วน

(4) ในส่วนของการบันทึกข้อมูลการใช้พลังงานไฟฟ้าแบบรวม

(4.1) ด้านล่าง DATASOURCE คลิกที่ ADD จะปรากฏ Datasource Type ชนิดต่างๆ ให้เลือกเป็น NETPIE Microgear

(4.2) ใส่ข้อมูลสำหรับ Datasource

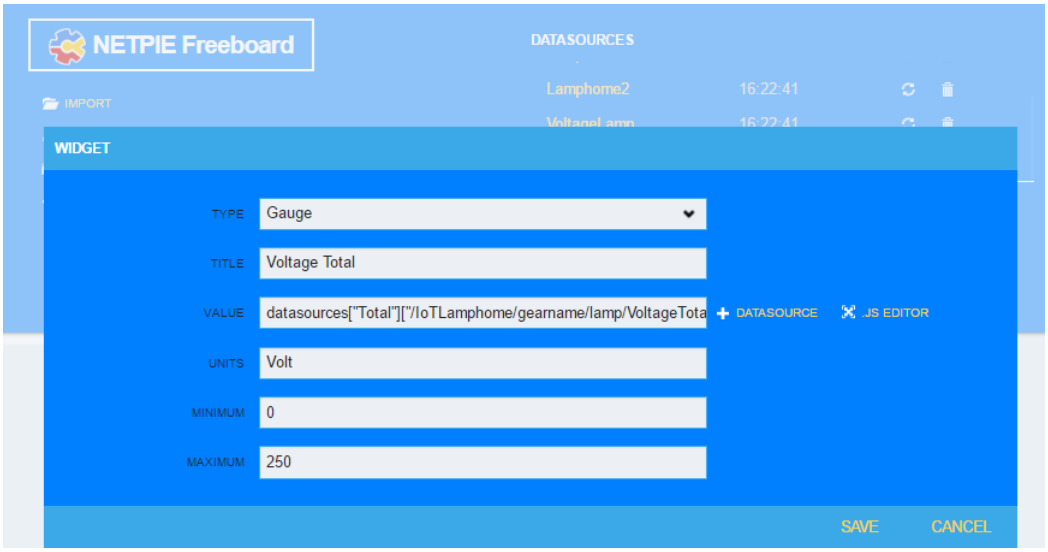


รูปที่ 3.26 การระบุข้อมูลของ Datasource สำหรับใช้ในการบันทึกข้อมูลการใช้พลังงานไฟฟ้าแบบรวม

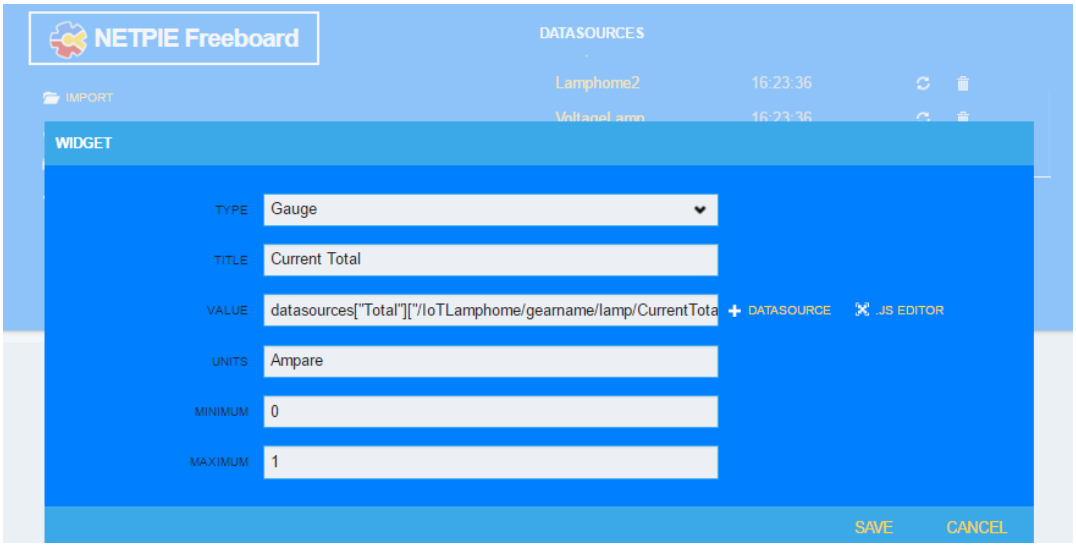
(4.3) เพิ่ม Panel สำหรับสร้าง Widget ด้วยการคลิก ADD PANE จะปรากฏ Panel เพิ่มขึ้นมาด้านล่าง

(4.4) เพิ่ม Widget บน Panel ที่สร้างขึ้นใหม่โดย คลิกที่เครื่องหมาย + และเลือกชนิดของ Widget โดยเลือกเป็น Gauge

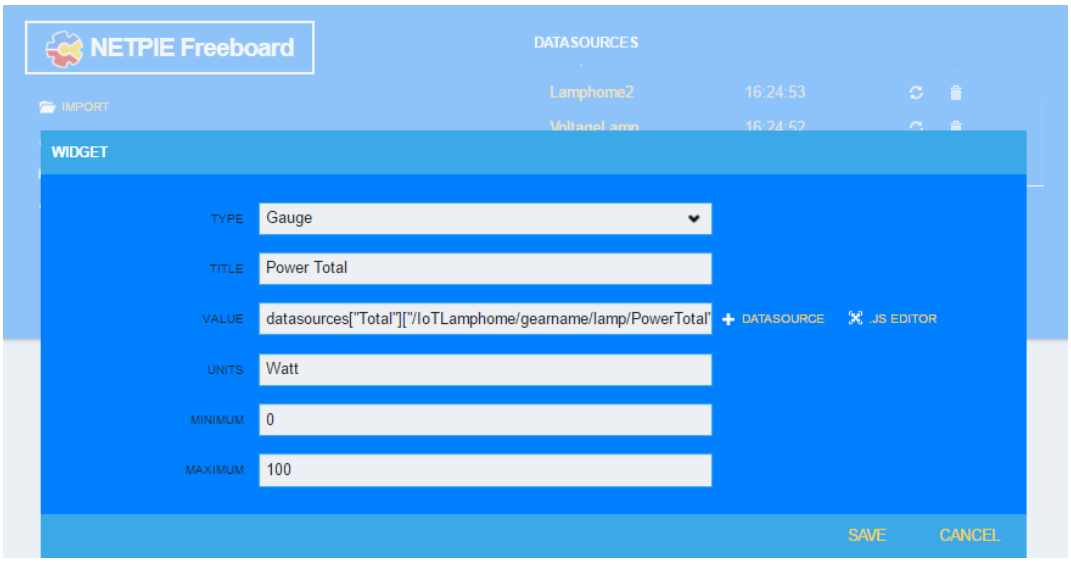
(4.5) ใส่ข้อมูลสำหรับ Widget



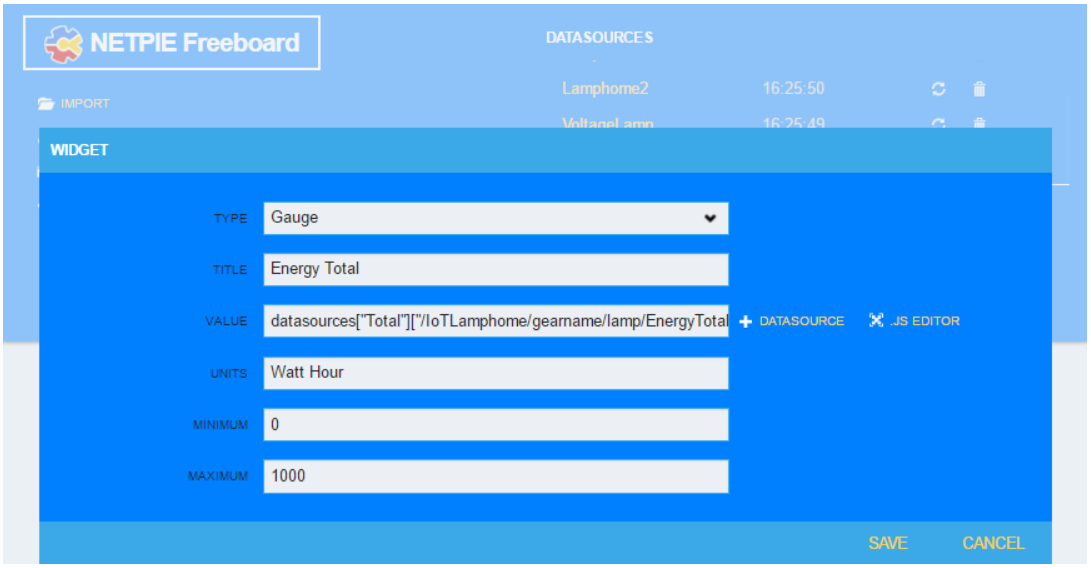
รูปที่ 3.27 การระบุข้อมูลของ Widget สำหรับสร้างการแสดงผลค่าแรงดันไฟฟ้าที่วัดแบบรวม



รูปที่ 3.28 การระบุข้อมูลของ Widget สำหรับสร้างการแสดงผลค่ากระแสไฟฟ้าที่วัดแบบรวม



รูปที่ 3.29 การระบุข้อมูลของ Widget สำหรับสร้างการแสดงผลค่ากำลังไฟฟ้าที่วัดแบบรวม



รูปที่ 3.30 การระบุข้อมูลของ Widget สำหรับสร้างการแสดงผลค่าพลังงานไฟฟ้าที่วัดแบบรวม

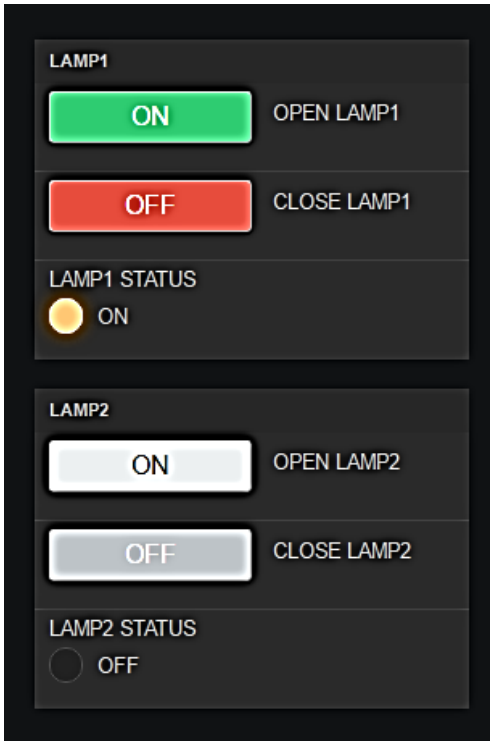
บทที่ 4
ผลการดำเนินงาน

ในบทนี้เป็นผลการทดลองตัวควบคุมอุปกรณ์ไฟฟ้าและบันทึกข้อมูลการใช้พลังงาน โดยควบคุมผ่านการเชื่อมต่ออินเตอร์เน็ต ทำให้สามารถควบคุม ปิด เปิด หลอดไฟ ได้จากทั้งเว็บฟรีบอร์ดของเน็ตพายหรือจากแอพพลิเคชั่นบนระบบปฏิบัติการแอนดรอยด์ รวมไปถึงการแสดงผลข้อมูลการใช้พลังงาน

4.1 การแสดงผลในส่วนของฟรีบอร์ดของเน็ตพาย

4.1.1 ตัวควบคุมอุปกรณ์ไฟฟ้า

ในหน้าฟรีบอร์ดจะมีการแสดง Widget สำหรับทำการควบคุมอุปกรณ์ไฟฟ้า โดยประกอบด้วย 2 Widget โดย Widget ที่ 1 มีชื่อว่า LAMP1 ใช้สำหรับทำการควบคุม ปิด เปิด หลอดไฟดวงที่ 1 และ Widget ที่ 2 มีชื่อว่า LAMP2 ใช้สำหรับทำการควบคุม ปิด เปิด หลอดไฟดวงที่ 2

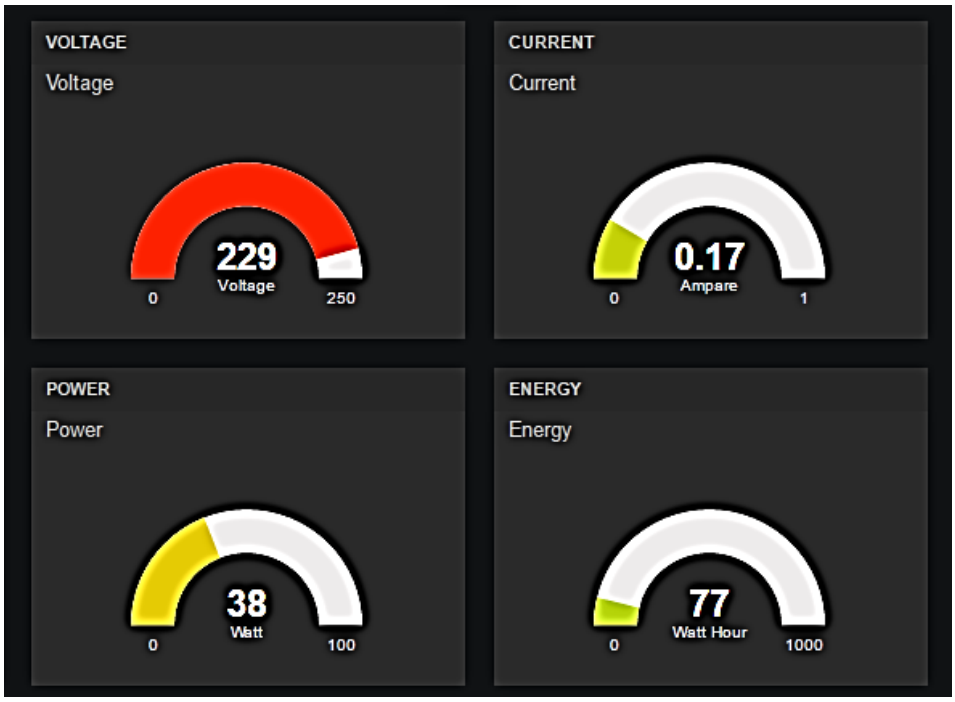


รูปที่ 4.1 Widget สำหรับทำการควบคุมหลอดไฟ

โดยที่ Widget ที่มีชื่อว่า LAMP1 มีการแสดงสถานะ ON และมีไฟสีส้มปรากฏให้เห็นแสดงให้เห็นถึงการทำงานของหลอดไฟ ว่าหลอดไฟดวงที่ 1 ได้มีการเปิดใช้งานอยู่ ส่วน Widget ที่มีชื่อว่า LAMP2 มีการแสดงสถานะ OFF และไม่มีไฟสีส้มปรากฏให้เห็นแสดงให้เห็นถึงการทำงานของหลอดไฟว่า หลอดไฟดวงที่ 2 ไม่ได้มีการเปิดใช้งาน

4.1.2 การแสดงผลการบันทึกข้อมูลการใช้พลังงาน

4.1.2.1 การบันทึกข้อมูลการใช้พลังงานแบบแยกส่วน



รูปที่ 4.2 Widget แสดงค่าการใช้พลังงานไฟฟ้าแบบแยกส่วน โดยแบ่งเป็น แรงดันไฟฟ้า กระแสไฟฟ้า กำลังงานไฟฟ้า และพลังงานไฟฟ้า

ในหน้าฟรีบอร์ดจะมีการแสดง Widget สำหรับทำการบันทึกข้อมูลการใช้พลังงานแบบแยกส่วน โดยประกอบด้วย 4 Widget โดย Widget ที่ 1 มีชื่อว่า VOLTAGE ใช้สำหรับทำการบันทึกค่าแรงดันไฟฟ้าที่ใช้ โดยมีหน่วยเป็นโวลต์ และสามารถแสดงผลได้ตั้งแต่ค่า 0–250 โวลต์ Widget ที่ 2 มีชื่อว่า CURRENT ใช้สำหรับทำการบันทึกค่ากระแสไฟฟ้าที่ใช้ โดยมีหน่วยเป็นแอมแปร์ และสามารถแสดงผลได้ตั้งแต่ค่า 0–1 แอมแปร์ Widget ที่ 3 มีชื่อว่า POWER ใช้สำหรับทำการบันทึกค่ากำลังงานไฟฟ้าที่ใช้ โดยมีหน่วยเป็นวัตต์ และสามารถแสดงผลได้ตั้งแต่ค่า 0–100 วัตต์ Widget ที่ 4 มีชื่อว่า ENERGY ใช้สำหรับทำการบันทึกค่าพลังงานไฟฟ้าที่ใช้ โดยมีหน่วยเป็นวัตต์ต่อชั่วโมง และสามารถแสดงผลได้ตั้งแต่ค่า 0–1000 วัตต์ต่อชั่วโมง

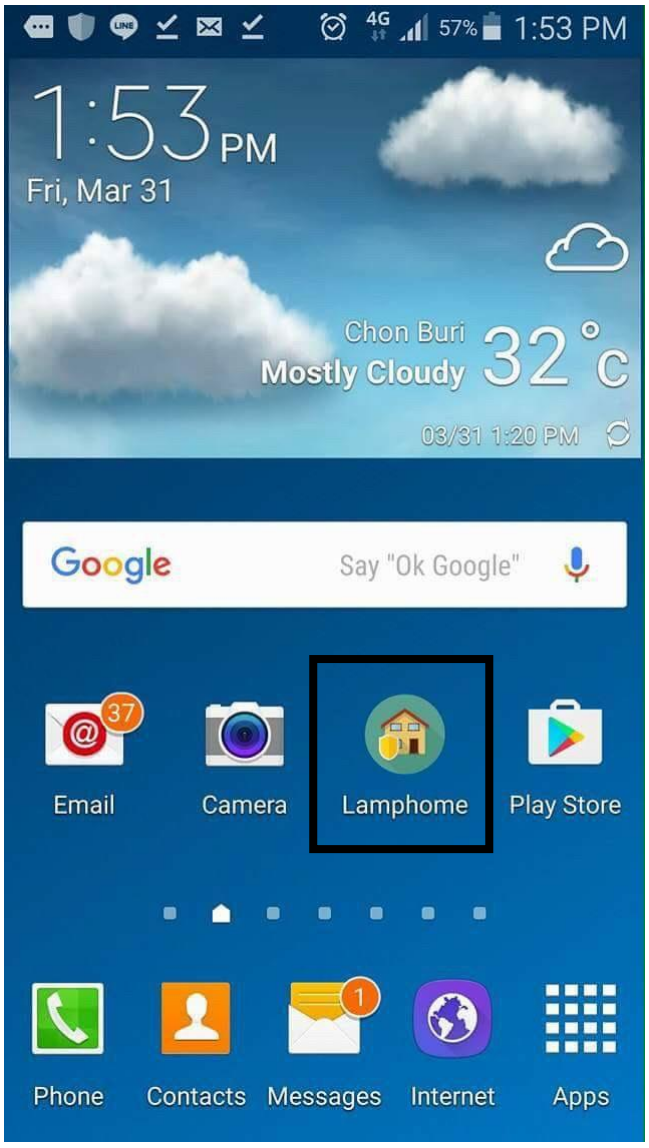
4.1.2.2 การบันทึกข้อมูลการใช้พลังงานโดยรวม

รูปที่ 4.3 Widget แสดงค่าการใช้พลังงานไฟฟ้าแบบรวมโดยแบ่งเป็น แรงดันไฟฟ้า กระแสไฟฟ้า กำลังงานไฟฟ้า และพลังงานไฟฟ้า

ในหน้าฟรีบอร์ดจะมีการแสดง Widget สำหรับทำการบันทึกข้อมูลการใช้พลังงานโดยรวม โดยประกอบด้วย 4 Widget โดย Widget ที่ 1 มีชื่อว่า VOLTAGETOTAL ใช้สำหรับทำการบันทึกค่าแรงดันไฟฟ้าที่ใช้ โดยมีหน่วยเป็นโวลต์ และสามารถแสดงผลได้ตั้งแต่ค่า 0–250 โวลต์ Widget ที่ 2 มีชื่อว่า CURRENTTOTAL ใช้สำหรับทำการบันทึกค่ากระแสไฟฟ้าที่ใช้ โดยมีหน่วยเป็นแอมแปร์ และสามารถแสดงผลได้ตั้งแต่ค่า 0–1 แอมแปร์ Widget ที่ 3 มีชื่อว่า POWERTOTAL ใช้สำหรับทำการบันทึกค่ากำลังงานไฟฟ้าที่ใช้ โดยมีหน่วยเป็นวัตต์ และสามารถแสดงผลได้ตั้งแต่ค่า 0–100 วัตต์ Widget ที่ 4 มีชื่อว่า ENERGYTTOTAL ใช้สำหรับทำการบันทึกค่าพลังงานไฟฟ้าที่ใช้ โดยมีหน่วยเป็นวัตต์ต่อชั่วโมง และสามารถแสดงผลได้ตั้งแต่ค่า 0–1000 วัตต์ต่อชั่วโมง

4.2 การแสดงผลในแอพพลิเคชั่นบนระบบปฏิบัติการแอนดรอยด์

หลังจากได้ออกแบบและพัฒนาแอพพลิเคชั่นบนระบบปฏิบัติการแอนดรอยด์ โดยทำการสร้างผ่าน MIT APPINVENTOR จะได้แอพพลิเคชั่น ดังรูปที่ 4.4



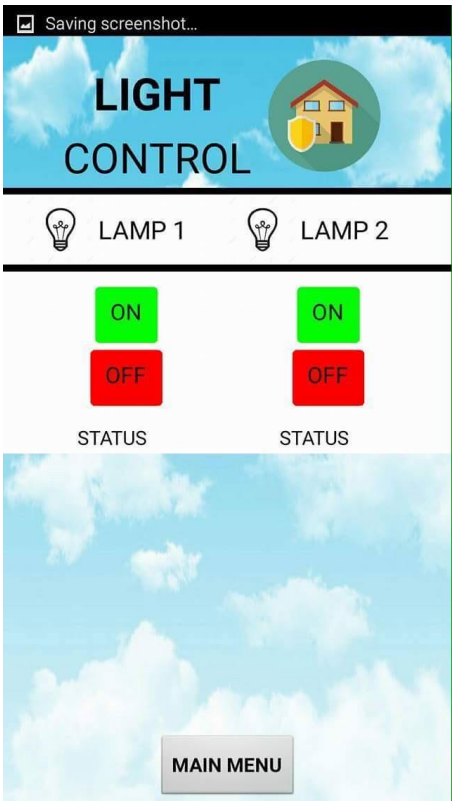
รูปที่ 4.4 หน้าตาแอพพลิเคชั่น ชื่อ Lamphome

ทำการคลิกไปที่ไอคอนของแอพพลิเคชั่น Lamphome บนสมาร์ทโฟน จะปรากฏหน้าจอภายในแอพพลิเคชั่นขึ้นมา โดยประกอบด้วยปุ่มกด 2 ปุ่ม คือ LIGHT CONTROL และ CONSUMPTION โดยที่ LIGHT CONTROL มีหน้าที่ทำการควบคุม ปิด เปิด หลอดไฟ และ CONSUMPTION มีหน้าที่ในการแสดงผลการใช้พลังงานของอุปกรณ์ไฟฟ้า

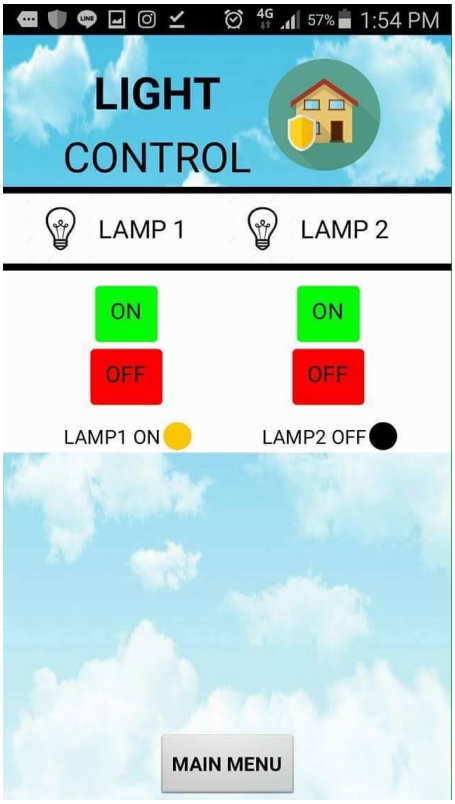
รูปที่ 4.5 หน้าจอหลักของแอพพลิเคชั่น Lamphome

ทำการคลิกไปที่ปุ่ม LIGHT CONTROL ในหน้าจอหลักของแอพพลิเคชั่น จะปรากฏหน้าจอ LIGHT CONTROL ขึ้นมา โดยประกอบด้วย 2 ส่วน คือ ส่วนการควบคุม ปิด เปิด หลอดไฟดวงที่ 1 ที่ชื่อว่า LAMP1 และส่วนควบคุม ปิด เปิด หลอดไฟดวงที่ 2 ที่ชื่อว่า LAMP 2 โดยในแต่ละส่วน จะปรากฏปุ่ม ON สีเขียวมีหน้าที่สำหรับกดเพื่อสั่งงานเปิดหลอดไฟ ปุ่ม OFF สีแดงมีหน้าที่สำหรับกดเพื่อสั่งงานปิดหลอดไฟ และภายใต้ปุ่ม ON OFF จะมีข้อความสำหรับแสดงสถานะ ปิด เปิด ของหลอดไฟ ส่วนด้านล่างสุดของหน้าจอ จะปรากฏปุ่มกด ที่มีคำว่า MAIN MENU กำกับอยู่ ทำหน้าที่ในการกลับไปที่หน้าจอหลักของแอพพลิเคชั่น ดังแสดงในรูปที่ 4.6

จะเห็นว่าเมื่อมีการสั่งให้หลอดไฟเปิดการใช้งาน ข้อความแสดงสถานะ จะแสดงคำว่า LAMP ON พร้อมสีแสดงสถานะเป็นสีเหลือง ดังที่แสดงใน LAMP1 และเมื่อมีการสั่งให้หลอดไฟปิดการใช้งาน ข้อความแสดงสถานะ จะแสดงคำว่า LAMP OFF พร้อมสีแสดงสถานะเป็นสีดำ ดังแสดงใน LAMP2 ดังแสดงในรูปที่ 4.7

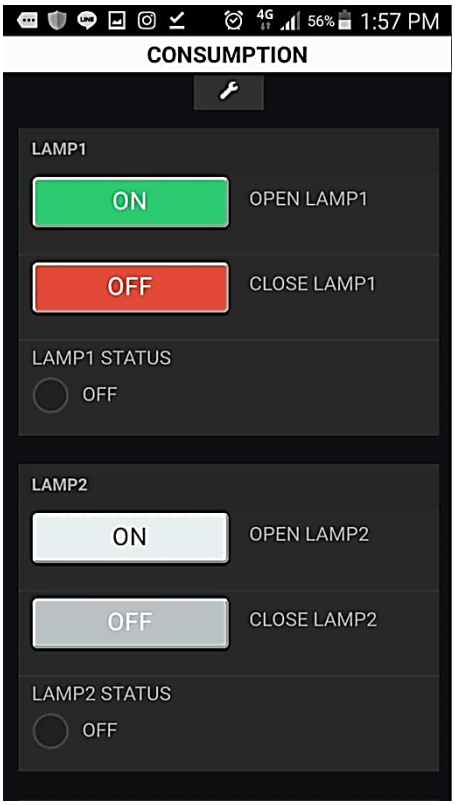


รูปที่ 4.6 หน้าจอ LIGHT CONTROL

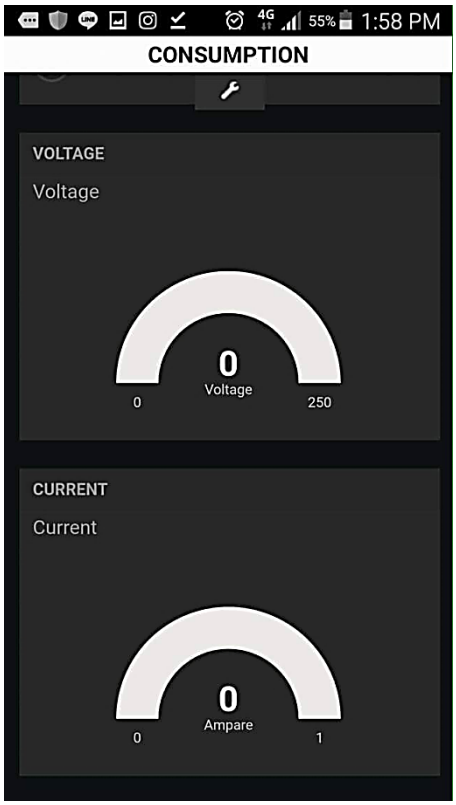


รูปที่ 4.7 หน้าจอแสดงสถานะของ LIGHT CONTROL

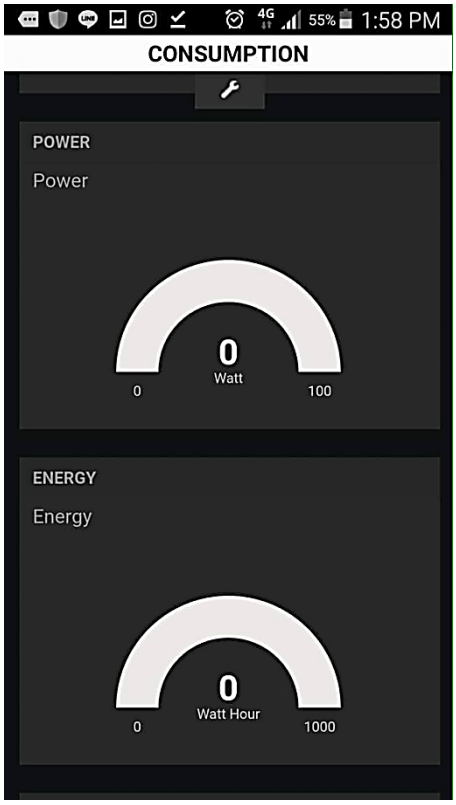
ทำการคลิกไปที่ปุ่ม CONSUMPTION ในหน้าจอหลักของแอพพลิเคชั่น จะปรากฏหน้าจอ CONSUMPTION ขึ้นมา พร้อมทั้งหน้าเว็บสำหรับล็อกอินเพื่อเข้าไปดูค่าการใช้พลังงานผ่านทางหน้าจอฟรีบอร์ดของเน็ตพาย โดยหลังจากทำการล็อกอินเข้าไปเรียบร้อยแล้ว จะปรากฏ Widget สำหรับทำการควบคุมปิดเปิดหลอดไฟ พร้อมทั้ง Widget แสดงค่า แรงดันไฟฟ้า กระแสไฟฟ้า กำลังงานไฟฟ้า และพลังงานไฟฟ้าที่ใช้



รูปที่ 4.8 หน้าจอควบคุม ปิด เปิดหลอดไฟ โดยผ่านทางฟรีบอร์ดของเน็ตพาย ภายในแอพพลิเคชั่น



รูปที่ 4.9 หน้าจอสำหรับแสดงผลแรงดันไฟฟ้าและกระแสไฟฟ้าที่ใช้



รูปที่ 4.10 หน้าจอสำหรับแสดงผลกำลังงานไฟฟ้าและพลังงานไฟฟ้าที่ใช้

4.3 ผลการดำเนินงานในส่วนของตัวควบคุมอุปกรณ์ไฟฟ้า

ตารางที่ 4.1 การควบคุมหลอดไฟดวงที่ 1 ผ่านเว็บฟรีบอร์ดของเน็ตพาย

ครั้งที่	ความสำเร็จในการสั่งงาน		ความล่าช้าในการทำงาน	
	เปิด	ปิด	เวลาเปิด (วินาที)	เวลาปิด (วินาที)
1	สำเร็จ	สำเร็จ	2.40	1.42
2	สำเร็จ	สำเร็จ	6.46	0.85
3	ไม่สำเร็จ	ไม่สำเร็จ	-	-
4	ไม่สำเร็จ	ไม่สำเร็จ	-	-
5	สำเร็จ	สำเร็จ	0.89	0.60
6	ไม่สำเร็จ	ไม่สำเร็จ	-	-
7	สำเร็จ	สำเร็จ	2.10	0.80
8	สำเร็จ	สำเร็จ	1.15	3.30
9	สำเร็จ	สำเร็จ	4.12	2.29
10	สำเร็จ	สำเร็จ	3.52	4.76
11	สำเร็จ	สำเร็จ	6.59	3.59
12	สำเร็จ	สำเร็จ	4.65	4.29
13	สำเร็จ	สำเร็จ	4.43	1.00
14	สำเร็จ	สำเร็จ	2.56	1.35
15	สำเร็จ	สำเร็จ	3.93	2.12
16	สำเร็จ	สำเร็จ	2.69	0.90
17	สำเร็จ	สำเร็จ	5.69	3.46
18	สำเร็จ	สำเร็จ	3.06	5.96
19	สำเร็จ	สำเร็จ	0.49	2.06
20	สำเร็จ	สำเร็จ	1.19	4.13
21	สำเร็จ	สำเร็จ	3.96	4.76
22	สำเร็จ	สำเร็จ	4.62	2.40
23	สำเร็จ	สำเร็จ	1.79	4.66
24	ไม่สำเร็จ	ไม่สำเร็จ	-	-
25	สำเร็จ	ไม่สำเร็จ	3.42	-

จากผลการทดลองตารางที่ 4.1 การควบคุมหลอดไฟดวงที่ 1 ผ่านเว็บฟรีบอร์ดของเน็ตพาย จะพบว่าในการทดลองสั่งเปิดหลอดไฟดวงที่ 1 จำนวน 25 ครั้ง และสั่งปิดหลอดไฟดวงที่ 1 จำนวน 25 ครั้ง รวมทั้งหมดเป็นจำนวน 50 ครั้ง โดยในการทดลองสั่งเปิดหลอดไฟทั้งหมด 25 ครั้ง ประสบความสำเร็จในการสั่งเปิดหลอดไฟเป็นจำนวน 21 ครั้ง และในการทดลองสั่งปิดหลอดไฟทั้งหมดจำนวน 25 ครั้ง ประสบความสำเร็จในการสั่งปิดอุปกรณ์เป็นจำนวน 20 ครั้ง ดังนั้นในการทดลองควบคุมหลอดไฟทั้งหมดจำนวน 50 ครั้ง ประสบความสำเร็จในการสั่งงาน เป็นจำนวน 41 ครั้ง คิดเป็นความสำเร็จ 82 เปอร์เซ็นต์

นอกจากความสำเร็จในการสั่งงานแล้ว ได้มีการบันทึกค่าการล่าช้าในการทำงานเปิด ปิดหลอดไฟ โดยในการสั่งงานเปิดหลอดไฟจำนวน 25 ครั้ง จะมีค่าความล่าช้าในการติดของหลอดไฟเป็น 3.32 วินาที และการสั่งงานปิดหลอดไฟจำนวน 25 ครั้ง จะมีค่าความล่าช้าในการดับของหลอดไฟเป็น 2.74 วินาที ดังนั้น ในการทดลองควบคุมหลอดไฟทั้งหมดจำนวน 50 ครั้ง พบว่าเกิดความล่าช้าในการดำเนินงานเป็นเวลา 3.03 วินาที

ตารางที่ 4.2 การควบคุมหลอดไฟดวงที่ 2 ผ่านเว็บฟรีบอร์ดของเน็ตพาย

ครั้งที่	ความสำเร็จในการสั่งงาน		ความล่าช้าในการทำงาน	
	เปิด	ปิด	เวลาเปิด (วินาที)	เวลาปิด (วินาที)
1	สำเร็จ	สำเร็จ	1.35	1.13
2	สำเร็จ	สำเร็จ	6.45	4.79
3	สำเร็จ	สำเร็จ	1.84	0.89
4	สำเร็จ	สำเร็จ	0.71	4.36
5	สำเร็จ	สำเร็จ	2.95	3.16
6	สำเร็จ	สำเร็จ	3.24	2.09
7	ไม่สำเร็จ	ไม่สำเร็จ	-	-
8	สำเร็จ	สำเร็จ	2.84	4.71
9	สำเร็จ	สำเร็จ	1.36	2.94
10	สำเร็จ	สำเร็จ	0.89	1.28
11	สำเร็จ	สำเร็จ	3.07	3.13
12	สำเร็จ	ไม่สำเร็จ	2.98	-
13	สำเร็จ	สำเร็จ	5.05	6.38
14	สำเร็จ	สำเร็จ	0.50	2.89
15	สำเร็จ	สำเร็จ	3.99	4.80
16	สำเร็จ	สำเร็จ	4.52	1.00
17	สำเร็จ	สำเร็จ	1.75	5.88
18	สำเร็จ	สำเร็จ	0.40	1.50
19	สำเร็จ	สำเร็จ	5.12	4.26
20	สำเร็จ	สำเร็จ	2.96	3.72
21	สำเร็จ	สำเร็จ	2.77	1.89
22	สำเร็จ	สำเร็จ	3.77	4.94
23	สำเร็จ	สำเร็จ	5.75	5.98
24	สำเร็จ	สำเร็จ	0.88	0.58
25	สำเร็จ	สำเร็จ	3.98	5.28

จากผลการทดลองตารางที่ 4.2 การควบคุมหลอดไฟดวงที่ 2 ผ่านเว็บฟรีบอร์ดของเน็ตพาย จะพบว่าในการทดลองสั่งเปิดหลอดไฟดวงที่ 1 จำนวน 25 ครั้ง และสั่งปิดหลอดไฟดวงที่ 1 จำนวน 25 ครั้ง รวมทั้งหมดเป็นจำนวน 50 ครั้ง โดยในการทดลองสั่งเปิดหลอดไฟทั้งหมด 25 ครั้ง ประสบความสำเร็จในการสั่งเปิดหลอดไฟเป็นจำนวน 24 ครั้ง และในการทดลองสั่งปิดหลอดไฟทั้งหมดจำนวน 25 ครั้ง ประสบความสำเร็จในการสั่งปิดอุปกรณ์เป็นจำนวน 23 ครั้ง ดังนั้นในการทดลองควบคุมหลอดไฟทั้งหมดจำนวน 50 ครั้ง ประสบความสำเร็จในการสั่งงาน เป็นจำนวน 47 ครั้ง คิดเป็นความสำเร็จ 94 เปอร์เซ็นต์

นอกจากความสำเร็จในการสั่งงานแล้ว ได้มีการบันทึกค่าการล่าช้าในการทำงานเปิด ปิดหลอดไฟ โดยในการสั่งงานเปิดหลอดไฟจำนวน 25 ครั้ง จะมีค่าความล่าช้าในการติดของหลอดไฟเป็น 2.88 วินาที และการสั่งงานปิดหลอดไฟจำนวน 25 ครั้ง จะมีค่าความล่าช้าในการดับของหลอดไฟเป็น 3.37 วินาที ดังนั้น ในการทดลองควบคุมหลอดไฟทั้งหมดจำนวน 50 ครั้ง พบว่าเกิดความล่าช้าในการดำเนินงานเป็นเวลา 3.12 วินาที

ตารางที่ 4.3 การควบคุมหลอดไฟดวงที่ 1 ผ่านแอพพลิเคชั่น

ครั้งที่	ความสำเร็จในการสั่งงาน		ความล่าช้าในการทำงาน	
	เปิด	ปิด	เวลาเปิด (วินาที)	เวลาปิด (วินาที)
1	สำเร็จ	สำเร็จ	1.05	2.15
2	สำเร็จ	สำเร็จ	1.52	1.53
3	สำเร็จ	สำเร็จ	1.56	2.03
4	สำเร็จ	ไม่สำเร็จ	1.92	-
5	สำเร็จ	สำเร็จ	0.96	1.56
6	สำเร็จ	สำเร็จ	1.76	1.06
7	สำเร็จ	สำเร็จ	1.73	1.12
8	สำเร็จ	สำเร็จ	1.63	1.62
9	สำเร็จ	สำเร็จ	1.83	1.62
10	สำเร็จ	สำเร็จ	1.22	1.19
11	สำเร็จ	สำเร็จ	0.75	1.12
12	สำเร็จ	สำเร็จ	1.70	1.46
13	สำเร็จ	สำเร็จ	0.75	0.99
14	สำเร็จ	สำเร็จ	0.69	0.99
15	สำเร็จ	สำเร็จ	0.75	2.03
16	สำเร็จ	สำเร็จ	0.80	1.09
17	สำเร็จ	สำเร็จ	1.29	1.29
18	สำเร็จ	สำเร็จ	1.63	0.75
19	สำเร็จ	สำเร็จ	0.59	0.92
20	สำเร็จ	สำเร็จ	1.09	1.22
21	สำเร็จ	สำเร็จ	1.20	1.00
22	สำเร็จ	สำเร็จ	1.23	1.09
23	สำเร็จ	สำเร็จ	1.66	1.02
24	สำเร็จ	สำเร็จ	1.56	1.55
25	สำเร็จ	สำเร็จ	1.16	1.32

จากผลการทดลองตารางที่ 4.3 การควบคุมหลอดไฟดวงที่ 1 ผ่านแอพพลิเคชั่น จะพบว่าในการทดลองสั่งเปิดหลอดไฟดวงที่ 1 จำนวน 25 ครั้ง และสั่งปิดหลอดไฟดวงที่ 1 จำนวน 25 ครั้ง รวมทั้งหมดเป็นจำนวน 50 ครั้ง โดยในการทดลองสั่งเปิดหลอดไฟทั้งหมด 25 ครั้ง ประสบความสำเร็จในการสั่งเปิดหลอดไฟเป็นจำนวน 25 ครั้ง และในการทดลองสั่งปิดหลอดไฟทั้งหมดจำนวน 25 ครั้ง ประสบความสำเร็จในการสั่งปิดอุปกรณ์เป็นจำนวน 25 ครั้ง ดังนั้นในการทดลองควบคุมหลอดไฟทั้งหมดจำนวน 50 ครั้ง ประสบความสำเร็จในการสั่งงาน เป็นจำนวน 50 ครั้ง คิดเป็นความสำเร็จ 100 เปอร์เซ็นต์

นอกจากความสำเร็จในการสั่งงานแล้ว ได้มีการบันทึกค่าการล่าช้าในการทำงานเปิด ปิดหลอดไฟ โดยในการสั่งงานเปิดหลอดไฟจำนวน 25 ครั้ง จะมีค่าความล่าช้าในการติดของหลอดไฟเป็น 1.28 วินาที และการสั่งงานปิดหลอดไฟจำนวน 25 ครั้ง จะมีค่าความล่าช้าในการดับของหลอดไฟเป็น 1.27 วินาที ดังนั้น ในการทดลองควบคุมหลอดไฟทั้งหมดจำนวน 50 ครั้ง พบว่าเกิดความล่าช้าในการดำเนินงานเป็นเวลา 1.28 วินาที

ตารางที่ 4.4 การควบคุมหลอดไฟดวงที่ 2 ผ่านแอพพลิเคชั่น

ครั้งที่	ความสำเร็จในการสั่งงาน		ความล่าช้าในการทำงาน	
	เปิด	ปิด	เวลาเปิด (วินาที)	เวลาปิด (วินาที)
1	สำเร็จ	สำเร็จ	1.70	0.80
2	สำเร็จ	สำเร็จ	1.40	0.61
3	สำเร็จ	สำเร็จ	1.44	1.70
4	สำเร็จ	สำเร็จ	1.44	0.91
5	สำเร็จ	สำเร็จ	0.88	0.48
6	สำเร็จ	สำเร็จ	1.16	1.45
7	สำเร็จ	สำเร็จ	1.36	0.98
8	สำเร็จ	สำเร็จ	1.23	1.36
9	สำเร็จ	สำเร็จ	0.90	0.80
10	สำเร็จ	สำเร็จ	0.40	1.00
11	สำเร็จ	สำเร็จ	0.70	0.85
12	สำเร็จ	สำเร็จ	0.86	1.32
13	สำเร็จ	สำเร็จ	0.90	0.80
14	สำเร็จ	สำเร็จ	1.40	0.80
15	สำเร็จ	สำเร็จ	1.50	1.86
16	สำเร็จ	สำเร็จ	0.80	1.82
17	สำเร็จ	สำเร็จ	0.90	0.80
18	สำเร็จ	สำเร็จ	1.10	1.30
19	สำเร็จ	สำเร็จ	1.50	0.90
20	สำเร็จ	สำเร็จ	1.86	1.30
21	สำเร็จ	สำเร็จ	1.30	1.00
22	สำเร็จ	สำเร็จ	1.60	0.80
23	สำเร็จ	สำเร็จ	1.23	0.96
24	สำเร็จ	สำเร็จ	1.13	0.90
25	สำเร็จ	สำเร็จ	1.01	1.40

จากผลการทดลองตารางที่ 4.4 การควบคุมหลอดไฟดวงที่ 1 ผ่านแอพพลิเคชั่น จะพบว่าในการทดลองสั่งเปิดหลอดไฟดวงที่ 1 จำนวน 25 ครั้ง และสั่งปิดหลอดไฟดวงที่ 1 จำนวน 25 ครั้ง รวมทั้งหมดเป็นจำนวน 50 ครั้ง โดยในการทดลองสั่งเปิดหลอดไฟทั้งหมด 25 ครั้ง ประสบความสำเร็จในการสั่งเปิดหลอดไฟเป็นจำนวน 25 ครั้ง และในการทดลองสั่งปิดหลอดไฟทั้งหมดจำนวน 25 ครั้ง ประสบความสำเร็จในการสั่งปิดอุปกรณ์เป็นจำนวน 25 ครั้ง ดังนั้นในการทดลองควบคุมหลอดไฟทั้งหมดจำนวน 50 ครั้ง ประสบความสำเร็จในการสั่งงาน เป็นจำนวน 50 ครั้ง คิดเป็นความสำเร็จ 100 เปอร์เซ็นต์

นอกจากความสำเร็จในการสั่งงานแล้ว ได้มีการบันทึกค่าการล่าช้าในการทำงานเปิด ปิดหลอดไฟ โดยในการสั่งงานเปิดหลอดไฟจำนวน 25 ครั้ง จะมีค่าความล่าช้าในการติดของหลอดไฟเป็น 1.20 วินาที และการสั่งงานปิดหลอดไฟจำนวน 25 ครั้ง จะมีค่าความล่าช้าในการดับของหลอดไฟเป็น 1.10 วินาที ดังนั้น ในการทดลองควบคุมหลอดไฟทั้งหมดจำนวน 50 ครั้ง พบว่าเกิดความล่าช้าในการดำเนินงานเป็นเวลา 1.15 วินาที

4.4 ผลการดำเนินงานในส่วนของการบันทึกข้อมูลการใช้พลังงาน

ตารางที่ 4.5 การบันทึกข้อมูลการใช้ไฟฟ้าแบบแยกส่วนด้วย PZEM-004T

ระยะเวลา (นาที)	แรงดันไฟฟ้า (โวลต์)	กระแสไฟฟ้า (แอมแปร์)	กำลังงานไฟฟ้า (วัตต์)	พลังงานไฟฟ้า (วัตต์ต่อชั่วโมง)
0	223.70	0.30	33	158
10	224.00	0.29	36	163
20	223.90	0.27	34	175
30	224.70	0.20	28	184
40	224.40	0.17	18	190
50	224.80	0.19	20	195
60	225.80	0.15	13	204
70	226.50	0.15	15	206
80	226.70	0.15	14	209
90	226.80	0.14	12	211
100	226.90	0.15	14	213
110	227.70	0.15	15	216
120	227.10	0.15	15	219

จากตารางที่ 4.5 เป็นข้อมูลที่ได้จากการวัดการใช้ไฟฟ้าแบบแยกส่วนด้วยตัวตรวจรู้ PZEM-004T โดยอุปกรณ์ไฟฟ้าที่ใช้ในการทำการทดลองวัดค่าการใช้ไฟฟ้าแบบแยกส่วนครั้งนี้ คือ โน้ตบุ๊คยี่ห้อ Samsung 1 ตัว จากการทดลอง พบว่าแรงดันไฟฟ้าโดยเฉลี่ยที่ใช้ในระยะเวลา 2 ชั่วโมง มีค่าเท่ากับ 225.62 โวลต์ กระแสไฟฟ้าโดยเฉลี่ยที่ใช้ในระยะเวลา 2 ชั่วโมง มีค่าเท่ากับ 0.19 แอมแปร์ กำลังงานไฟฟ้าโดยเฉลี่ยที่ใช้ในระยะเวลา 2 ชั่วโมง มีค่าเท่ากับ 20 วัตต์ และค่าพลังงานไฟฟ้าที่ใช้งานมีค่าเท่ากับ 30 วัตต์ต่อชั่วโมง

ตารางที่ 4.6 การบันทึกข้อมูลการใช้ไฟฟ้าแบบแยกส่วนด้วย Multifunction Mini Ammeter

ระยะเวลา (นาที)	แรงดันไฟฟ้า (โวลต์)	กระแสไฟฟ้า (แอมแปร์)	กำลังงานไฟฟ้า (วัตต์)	พลังงานไฟฟ้า (วัตต์ต่อชั่วโมง)
0	223.80	0.21	47	-
10	224.10	0.22	49	-
20	223.50	0.20	55	-
30	225.10	0.19	43	-
40	224.20	0.13	29	-
50	224.40	0.14	31	-
60	225.00	0.11	25	-
70	226.10	0.13	29	-
80	226.80	0.12	27	-
90	226.90	0.11	25	-
100	227.00	0.11	25	-
110	227.30	0.11	25	-
120	227.10	0.12	27	-

จากตารางที่ 4.6 เป็นข้อมูลที่ได้จากการวัดการใช้ไฟฟ้าแบบแยกส่วนด้วยตัวตรวจรู้ Multifunction Mini Ammeter โดยอุปกรณ์ไฟฟ้าที่ใช้ในการทำการทดลองวัดค่าการใช้ไฟฟ้าแบบแยกส่วนครั้งนี้ คือ โน้ตบุ๊คยี่ห้อ Samsung 1 ตัว จากการทดลอง พบว่าแรงดันไฟฟ้าโดยเฉลี่ยที่ใช้ในระยะเวลา 2 ชั่วโมง มีค่าเท่ากับ 225.48 โวลต์ กระแสไฟฟ้าโดยเฉลี่ยที่ใช้ในระยะเวลา 2 ชั่วโมง มีค่าเท่ากับ 0.15 แอมแปร์ กำลังงานไฟฟ้าโดยเฉลี่ยที่ใช้ในระยะเวลา 2 ชั่วโมง มีค่าเท่ากับ 34 วัตต์

ซึ่งเมื่อนำค่าที่ได้จากการทดลองมาเปรียบเทียบกับค่าจริง จะพบว่าค่าแรงดันไฟฟ้าที่ได้จากการทดลองมีค่าความคลาดเคลื่อนเป็น 0.06 เปอร์เซ็นต์ ค่ากระแสไฟฟ้าที่ได้จากการทดลองมีค่าความคลาดเคลื่อน เป็น 26.67 เปอร์เซ็นต์ ค่ากำลังงานไฟฟ้าที่ได้จากการทดลองมีค่าความคลาดเคลื่อน เป็น 11.76 เปอร์เซ็นต์

ตารางที่ 4.7 การบันทึกข้อมูลการใช้ไฟฟ้าโดยรวม ด้วย PZEM-004T

ระยะเวลา (นาที)	แรงดันไฟฟ้า (โวลต์)	กระแสไฟฟ้า (แอมแปร์)	กำลังงานไฟฟ้า (วัตต์)	พลังงานไฟฟ้า (วัตต์ต่อชั่วโมง)
0	225.40	0.12	18	220
10	226.00	0.11	15	225
20	225.50	0.12	16	233
30	225.90	0.13	15	237
40	225.80	0.11	14	240
50	226.00	0.10	16	244
60	225.50	0.09	15	249
70	225.20	0.08	12	253
80	225.10	0.07	14	257
90	225.90	0.07	13	260
100	225.00	0.07	13	262
110	225.50	0.07	12	265
120	225.80	0.07	14	269

จากตารางที่ 4.7 เป็นข้อมูลที่ได้จากการวัดการใช้ไฟฟ้าแบบรวมด้วยตัวตรวจรู้ PZEM-004T โดยอุปกรณ์ไฟฟ้าที่ใช้ในการทำการทดลองวัดค่าการใช้ไฟฟ้าแบบรวมครั้งนี้ประกอบด้วยอุปกรณ์ไฟฟ้า 2 ตัวด้วยกัน คือ โทรศัพท์ยี่ห้อ Samsung 1 เครื่อง และโคมไฟ 1 โคม จากการทดลอง พบว่าแรงดันไฟฟ้าโดยเฉลี่ยที่ใช้ในระยะเวลา 2 ชั่วโมง มีค่าเท่ากับ 225.94 โวลต์ กระแสไฟฟ้าโดยเฉลี่ยที่ใช้ในระยะเวลา 2 ชั่วโมง มีค่าเท่ากับ 0.09 แอมแปร์ กำลังงานไฟฟ้าโดยเฉลี่ยที่ใช้ในระยะเวลา 2 ชั่วโมง มีค่าเท่ากับ 14 วัตต์ และค่าพลังงานไฟฟ้าที่ใช้งานมีค่าเท่ากับ 24 วัตต์ต่อชั่วโมง

ตารางที่ 4.8 การบันทึกข้อมูลการใช้ไฟฟ้าโดยรวม ด้วย Multifunction Mini Ammeter

ระยะเวลา (นาที)	แรงดันไฟฟ้า (โวลต์)	กระแสไฟฟ้า (แอมแปร์)	กำลังงานไฟฟ้า (วัตต์)	พลังงานไฟฟ้า (วัตต์ต่อชั่วโมง)
0	225.70	0.08	18	-
10	227.00	0.07	15	-
20	227.00	0.07	17	-
30	226.70	0.07	16	-
40	227.10	0.07	15	-
50	227.50	0.08	18	-
60	227.90	0.07	16	-
70	228.10	0.06	14	-
80	228.20	0.06	15	-
90	228.00	0.06	14	-
100	227.20	0.06	14	-
110	226.30	0.06	14	-
120	225.80	0.06	14	-

จากตารางที่ 4.8 เป็นข้อมูลที่ได้จากการวัดการใช้ไฟฟ้าแบบรวมด้วยตัวตรวจรู้ Multifunction Mini Ammeter โดยอุปกรณ์ไฟฟ้าที่ใช้ในการทำการทดลองวัดค่าการใช้ไฟฟ้าแบบรวมครั้งนี้ประกอบด้วยอุปกรณ์ไฟฟ้า 2 ตัวด้วยกัน คือ โทรศัพท์ยี่ห้อ Samsung 1 เครื่อง และโคมไฟ 1 โคม จากการทดลองพบว่าแรงดันไฟฟ้าโดยเฉลี่ยที่ใช้ในระยะเวลา 2 ชั่วโมง มีค่าเท่ากับ 227.12 โวลต์ กระแสไฟฟ้าโดยเฉลี่ยที่ใช้ในระยะเวลา 2 ชั่วโมง มีค่าเท่ากับ 0.07 แอมแปร์ กำลังงานไฟฟ้าโดยเฉลี่ยที่ใช้ในระยะเวลา 2 ชั่วโมง มีค่าเท่ากับ 15 วัตต์

ซึ่งเมื่อนำค่าที่ได้จากการทดลองมาเปรียบเทียบกับค่าจริง จะพบว่า ค่าแรงดันไฟฟ้าที่ได้จากการทดลองมีค่าความคลาดเคลื่อน เป็น 0.52 เปอร์เซ็นต์ ค่ากระแสไฟฟ้าที่ได้จากการทดลองมีค่าความคลาดเคลื่อน เป็น 28.57 เปอร์เซ็นต์ ค่ากำลังงานไฟฟ้าที่ได้จากการทดลองมีค่าความคลาดเคลื่อน เป็น 6.67 เปอร์เซ็นต์

บทที่ 5
สรุปผล และข้อเสนอแนะ

5.1 สรุปผล

จากการทำการทดลองควบคุม เปิด ปิด หลอดไฟ จากฟรีบอร์ดของเน็ตพายและแอพพลิเคชั่นบนสมาร์ทโฟนของระบบปฏิบัติการแอนดรอยด์ผ่านอินเตอร์เน็ต โดยทำการทดลองกับหลอดไฟเป็นจำนวน 2 หลอด พบว่า การทดลองควบคุมหลอดไฟดวงที่ 1 ผ่านฟรีบอร์ดของเน็ตพาย มีเปอร์เซ็นต์ความสำเร็จในการทดลองเท่ากับ 82 เปอร์เซ็นต์ และเกิดความล่าช้าในการดำเนินการเป็นเวลา 3.03 วินาที การทดลองควบคุมหลอดไฟดวงที่ 2 ผ่านฟรีบอร์ดของเน็ตพาย มีเปอร์เซ็นต์ความสำเร็จในการทดลองเท่ากับ 94 เปอร์เซ็นต์ และเกิดความล่าช้าในการดำเนินการเป็นเวลา 3.12 วินาที การทดลองควบคุมหลอดไฟดวงที่ 1 ผ่านแอพพลิเคชั่นบนสมาร์ทโฟน มีเปอร์เซ็นต์ความสำเร็จในการทดลองเท่ากับ 100 เปอร์เซ็นต์ และเกิดความล่าช้าในการดำเนินการเป็นเวลา 1.28 วินาที การทดลองควบคุมหลอดไฟดวงที่ 2 ผ่านแอพพลิเคชั่นบนสมาร์ทโฟน มีเปอร์เซ็นต์ความสำเร็จในการทดลองเท่ากับ 100 เปอร์เซ็นต์ และเกิดความล่าช้าในการดำเนินการเป็นเวลา 1.15 วินาที จึงสรุปได้ว่าสามารถทำการ เปิด ปิด อุปกรณ์ไฟฟ้าผ่านฟรีบอร์ดของเน็ตพายและผ่านแอพพลิเคชั่นบนสมาร์ทโฟนได้ โดยที่การสั่งเปิด ปิด อุปกรณ์ไฟฟ้าผ่านแอพพลิเคชั่นมีเปอร์เซ็นต์ความสำเร็จในการเปิด ปิด ที่สูงกว่าและมีความล่าช้าน้อยกว่าการสั่งเปิด ปิด อุปกรณ์ไฟฟ้าผ่านทางฟรีบอร์ดของเน็ตพาย อาจเนื่องมาจากความผิดพลาดจากการทำการกดปุ่มเปิด ปิด หรือความล่าช้าของเครือข่ายอินเตอร์เน็ต

ส่วนการทดลองบันทึกและแสดงผลการใช้พลังงานของอุปกรณ์ไฟฟ้า เราได้แบ่งออกเป็นสองรูปแบบคือ การวัดการใช้พลังงานไฟฟ้าแบบแยกส่วน และการวัดการใช้พลังงานไฟฟ้าโดยรวม โดยได้ทำการทดลองเพื่อเปรียบเทียบค่าจริงที่ระบุไว้บนตัวอุปกรณ์ไฟฟ้า จากการทดลองพบว่า เมื่อทำการเปรียบเทียบค่าที่ได้จากการทดลองที่ใช้ตัวตรวจรู้ PZEM-004T และค่าจริงของการวัดแบบแยกส่วน มีค่าความคลาดเคลื่อนของการวัดค่าแรงดันไฟฟ้าจากการทดลองอยู่ที่ 0.06 เปอร์เซ็นต์ มีค่าความคลาดเคลื่อนของการวัดค่ากระแสไฟฟ้าจากการทดลองอยู่ที่ 26.67 เปอร์เซ็นต์ มีค่าความคลาดเคลื่อนของการวัดค่าพลังไฟฟ้าจากการทดลองอยู่ที่ 11.76 เปอร์เซ็นต์ และเมื่อทำการเปรียบเทียบค่าที่ได้จากการทดลองที่ใช้ตัวตรวจรู้ PZEM-004T และค่าจริงของการวัดโดยรวม มีค่าความคลาดเคลื่อนของการวัดค่าแรงดันไฟฟ้าจากการทดลองอยู่ที่ 0.52 เปอร์เซ็นต์ มีค่าความคลาดเคลื่อนของการวัดค่ากระแสไฟฟ้าจากการทดลองอยู่ที่ 28.57 เปอร์เซ็นต์ มีค่าความคลาดเคลื่อนของการวัดค่าพลังไฟฟ้าจากการทดลองอยู่ที่ 6.67 เปอร์เซ็นต์ ซึ่งจากความคลาดเคลื่อนดังกล่าว ไม่ได้สูงจนเกินไป ดังนั้นจึงสามารถนำตัวตรวจรู้ PZEM-004T ไปใช้งานในการทำการบันทึกค่าการใช้พลังงานได้

5.2 ข้อเสนอแนะ

จากการทำการทดลองควรมีการศึกษาข้อมูลของตัวอุปกรณ์ในส่วนของฮาร์ดแวร์ ให้ละเอียดรอบคอบ เนื่องจากการทดลองเป็นการทำงานกับอุปกรณ์ที่ใช้ไฟฟ้าแรงดัน 220 โวลต์ ซึ่งอาจทำให้เกิดอันตรายได้ ควรมีการเลือกบอร์ดไมโครคอนโทรลเลอร์ที่จะใช้ในการทดลองให้ตรงกับคลาวด์ที่ใช้ในการรับส่งข้อมูล และความล่าช้าของการรับส่งข้อมูลอาจขึ้นอยู่กับความเร็วของอินเตอร์เน็ตที่ใช้ในการเชื่อมต่อ

เอกสารอ้างอิง

NECTEC a member of NSTDA. คู่มือการใช้งาน NETPIE. 2559.

ไพโรจน์ ไววานิชกิจ. Cloud Computing หัวใจโลกธุรกิจยุคใหม่. เซมิคอนดักเตอร์ อิเล็กทรอนิกส์. 2558:101-6.

ประพนธ์ อัศวภาณุวัฒน์. เรียนให้รู้ เล่นให้เป็น ใช้ให้เกิดประโยชน์ ตอน MIT App Inventor กับการพัฒนาโปรแกรมแอนดรอยด์. ไมโครคอมพิวเตอร์. 2555:163.

ปัญญา มัฆะศร. หลักการพื้นฐานการเชื่อมต่ออุปกรณ์ Internet of Things (ตอนที่ 1). Modern Manufacturing. 2559:53.

พงศ์พันธุ์ ปริยวงศ์. การวิเคราะห์การเชื่อมโยงในระบบ IoT. เซมิคอนดักเตอร์ อิเล็กทรอนิกส์. 2559: 80-2.

มงคล พรหมเทศ. งานไฟฟ้าทั่วไป. กรุงเทพมหานคร: เอมพันธุ์; 2542.

รุ่งโรจน์ เกื้อกูลพงศ์. วิวัฒนาการของระบบปฏิบัติการแอนดรอยด์ (Android) สู่เทคโนโลยีในอนาคต. เซมิคอนดักเตอร์ อิเล็กทรอนิกส์. 2556:135-7.

ประวัติย่อผู้ทำโครงงาน

ประวัติย่อผู้ทำโครงงาน

ชื่อ ชื่อสกุล	นายธนวิชญ์ กมลฉ่ำ
วันเดือนปีเกิด	13 สิงหาคม 2537
สถานที่เกิด	เขตป้อมปราบ จังหวัดกรุงเทพมหานคร
สถานที่อยู่ปัจจุบัน	27/4 หมู่ 7 หมู่บ้านจิตรธารา
	ถนนพุทธมณฑลสาย 3 เขตทวีวัฒนา
	แขวงทวีวัฒนา กรุงเทพมหานคร 10170
หมายเลขโทรศัพท์ติดต่อ	092-280-4921

ประวัติการศึกษา

พ.ศ. 2556	มัธยมศึกษาปีที่ 6
	จากโรงเรียนอัสสัมชัญ ธนบุรี
พ.ศ. 2560	กำลังศึกษาระดับปริญญาตรี สาขาวิชาวิศวกรรมไฟฟ้า
	คณะวิศวกรรมศาสตร์ มหาวิทยาลัยศรีนครินทรวิโรฒ

ประวัติย่อผู้ทำโครงงาน

ชื่อ ชื่อสกุล	นางสาวหทัยรัตน์ พินิจสุวรรณ
วันเดือนปีเกิด	23 มกราคม 2538
สถานที่เกิด	อำเภอเมือง จังหวัดแพร่
สถานที่อยู่ปัจจุบัน	425 หมู่ 9 ต.นาปรัง อ.ปง
	จ.พะเยา 56140
หมายเลขโทรศัพท์ติดต่อ	087-248-7122

ประวัติการศึกษา

พ.ศ. 2556	มัธยมศึกษาปีที่ 6
	จากโรงเรียนสาธิตมหาวิทยาลัยเชียงใหม่
พ.ศ. 2560	กำลังศึกษาระดับปริญญาตรี สาขาวิชาวิศวกรรมไฟฟ้า
	คณะวิศวกรรมศาสตร์ มหาวิทยาลัยศรีนครินทรวิโรฒ