

ระบบบันทึกปริมาณกระแสไฟฟ้าผ่านระบบเครือข่ายแบบไร้สาย
THE CURRENT LOGGING SYSTEM WITH WIRELESS LAN



ปริญญาานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมการวัดคุม
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2560

ระบบบันทึกปริมาณกระแสไฟฟ้าผ่านระบบเครือข่ายแบบไร้สาย
THE CURRENT LOGGING SYSTEM WITH WIRELESS LAN



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมการวัดคุม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

THE CURRENT LOGGING SYSTEM WITH WIRELESS LAN



A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENT FOR THE DEGREE OF
BACHELOR OF ENGINEERING IN INSTRUMENTATION ENGINEERING
FACULTY OF ENGINEERING
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการ **ACADEMIC YEAR 2017** ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาานิพนธ์ปีการศึกษา 2560
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ใบรับรองปริญญาานิพนธ์

หัวข้อปริญญาานิพนธ์ ระบบบันทึกปริมาณกระแสไฟฟ้าผ่านระบบเครือข่ายแบบไร้สาย
THE CURRENT LOGGING SYSTEM WITH WIRELESS LAN
นักศึกษาผู้จัดทำ นางสาวชญัญญาช เฒ่าอุดม รหัสนักศึกษา 57010296
นายพิษณุพัฒน์ สุวรรณสิงห์ รหัสนักศึกษา 57010909
ปริญญา วิศวกรรมศาสตรบัณฑิต
สาขาวิชา วิศวกรรมการวัดคุม
ปีการศึกษา 2560

อาจารย์ผู้ควบคุมปริญญาานิพนธ์	ลายมือชื่อ
ผู้ช่วยศาสตราจารย์สุธรรม สัทธรรมสกุล	
รองศาสตราจารย์ ดร.วิศรุต ศรีรัตนะ	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	ระบบบันทึกปริมาณกระแสไฟฟ้าผ่านระบบเครือข่ายแบบไร้สาย		
	THE CURRENT LOGGING SYSTEM WITH WIRELESS LAN		
นักศึกษาผู้จัดทำ	นางสาวชญานุช	เฒ่าอุดม	รหัสนักศึกษา 57010296
	นายพิษณุพัฒน์	สุวรรณสิงห์	รหัสนักศึกษา 57010909
อาจารย์ที่ปรึกษา	ผู้ช่วยศาสตราจารย์สุธรรม		สีทธรรมสกุล
	รองศาสตราจารย์ ดร.วิศรุต ศรีรัตนะ		
ปีการศึกษา	2560		

บทคัดย่อ

โครงการนี้จัดทำขึ้นเพื่อศึกษาการสร้างอุปกรณ์เก็บข้อมูลปริมาณการใช้กระแสไฟฟ้า ของเครื่องปรับอากาศ แล้วนำข้อมูลที่ได้บันทึกเป็นฐานข้อมูลไว้บนเครื่องแม่ข่าย (Server) เพื่อนำไปใช้ในการตรวจสอบปริมาณการใช้พลังงานของเครื่องปรับอากาศ หรืออาจนำไปใช้ในการวิเคราะห์หาสาเหตุความผิดปกติของเครื่องปรับอากาศได้ โดยมีอุปกรณ์หลัก ประกอบด้วยโมดูลเซนเซอร์ 3 โมดูล คือ โมดูลเซนเซอร์วัดกระแสไฟฟ้า (ACS712-20A) , โมดูลเซนเซอร์วัดแรงดันไฟฟ้า (ZMPT101B), โมดูลเซนเซอร์วัดอุณหภูมิ (AM2302) พร้อมทั้งโมดูลนาฬิกา (Real Time Clock DS3231) และไมโครคอนโทรลเลอร์พร้อมเครือข่ายแบบไร้สาย (NodeMCU V3) ซึ่งจะทำหน้าที่รับค่าจากโมดูลเซนเซอร์ เพื่อส่งค่าดังกล่าวไปบันทึกลงบนฐานข้อมูลบนเครื่องแม่ข่ายผ่านระบบอินเทอร์เน็ต และผู้ใช้งานสามารถเลือกดูข้อมูลรายงานย้อนหลังในลักษณะต่าง ๆ ได้ผ่านทางหน้าเครื่องคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Thesis Title	THE CURRENT LOGGING SYSTEM WITH WIRELESS LAN	
Authors	Ms. Chyanut	Thao-udom
	Mr. Pitsanupat	Suwannasing
Thesis Advisor	Asst. Prof. Sutham	Satthamsakul
	Assoc. Prof. Dr. Witsarut	Sriratana
Year	2017	

ABSTRACT

This project presents a study and construction of the air-conditioner current logging system. The system comprises three sensor modules, namely the current sensor module (ACS712-20A), the voltage sensor module (ZMPT101B), the temperature sensor module (AM2302); a real time clock module (DS3231), and a microcontroller with wireless LAN (NodeMCU V3). Data from all sensor modules are received, processed via a microcontroller, and sent to the database over the internet. This feature also allows users to view the data logging online.

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สำเร็จลุล่วงได้ด้วยดี เนื่องจากได้รับความช่วยเหลือและดูแลเอาใจใส่ โดยเฉพาะอย่างยิ่งจากอาจารย์ที่ปรึกษา ผู้ช่วยศาสตราจารย์สุธรรม สัทธรรมสกุล และรองศาสตราจารย์ ดร.วิศรุต ศรีรัตน์ ในการแนะนำ ตรวจสอบแก้ไข ให้ข้อเสนอแนะ ติดตามความก้าวหน้าในการดำเนินการ ผู้จัดทำรู้สึกซาบซึ้งในความกรุณาของอาจารย์สองท่านนี้เป็นอย่างยิ่ง และขอขอบพระคุณเป็นอย่างสูงไว้ ณ โอกาสนี้

คณะผู้จัดทำ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
สารบัญตาราง.....	VII
สารบัญรูป.....	VIII

บทที่ 1 บทนำ.....	1
1.1 ความสำคัญของปริญญาโท.....	1
1.2 วัตถุประสงค์ของปริญญาโท.....	1
1.3 ขอบเขตของปริญญาโท.....	1
1.4 ขั้นตอนการศึกษา.....	1
1.5 ประโยชน์ที่คาดว่าจะได้รับ.....	2
บทที่ 2 ทฤษฎีและหลักการที่เกี่ยวข้อง.....	3
2.1 ไมโครคอนโทรลเลอร์.....	3
2.1.1 บอร์ด NodeMCU V3.....	3
2.1.2 คุณสมบัติของบอร์ด NodeMCU V3.....	3
2.1.3 ข้อมูลเชิงเทคนิคของบอร์ด NodeMCU V3.....	4
2.1.4 ไอซีเบอร์ MCP3008.....	5
2.2 โมดูลเซนเซอร์วัดกระแสไฟฟ้า ACS712-20A.....	5
2.3 โมดูลเซนเซอร์วัดแรงดันไฟฟ้า ZMPT 101B.....	6
2.4 โมดูลเซนเซอร์วัดอุณหภูมิและความชื้น AM2302.....	7
2.5 โมดูลนาฬิกา DS3231.....	8
2.6 ระบบจัดการฐานข้อมูล.....	10
2.6.1 หน้าที่ของระบบจัดการฐานข้อมูล.....	10
2.6.2 องค์ประกอบของระบบฐานข้อมูล.....	12
2.6.2.1 ฮาร์ดแวร์ (Hardware).....	12
2.6.2.2 โปรแกรม (Program).....	12

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา IV และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	หน้า
2.6.2.3 ข้อมูล (Data).....	12
2.6.2.4 บุคลากร (People).....	12
2.6.2.5 ขั้นตอนปฏิบัติงาน (Procedures).....	13
2.6.3 ข้อดีในการใช้ระบบฐานข้อมูล.....	13
2.6.4 ข้อเสียในการใช้ระบบฐานข้อมูล.....	14
2.6.5 โปรแกรมระบบจัดการฐานข้อมูล MySQL.....	15
2.6.6 โปรแกรม phpMyAdmin.....	15
2.7 เว็บเซิร์ฟเวอร์ (Web Server).....	16
2.8 โปรแกรม Text Editor.....	16
2.9 ภาษา C.....	17
2.9.1 โครงสร้างของโปรแกรมภาษา C.....	17
2.9.2 ตัวแปรในภาษา C.....	17
2.9.3 ฟังก์ชันในการรับและแสดงผลข้อมูล.....	18
2.9.3.1 ฟังก์ชันอื่น ๆ ที่ใช้ในการรับและแสดงข้อมูล.....	19
2.9.4 เครื่องหมายดำเนินการ (Operators).....	20
2.9.4.1 ตัวดำเนินการทางคณิตศาสตร์.....	20
2.9.4.2 ตัวดำเนินการสัมพันธ์และตัวดำเนินการตรรกะ.....	20
2.9.4.3 ตัวดำเนินการประกอบ.....	21
2.10 ภาษา PHP.....	22
2.11 ภาษา HTML.....	23
2.12 ภาษาจาวาสคริปต์.....	23
2.13 ไฟฟ้ากระแสสลับ.....	24
2.13.1 แรงดันไฟฟ้าและกระแสไฟฟ้า RMS.....	26
2.13.2 กำลังไฟฟ้าในระบบไฟฟ้ากระแสสลับ.....	26
2.13.2.1 การคำนวณกำลังไฟฟ้าจริง.....	26
2.13.2.2 การคำนวณกำลังไฟฟ้าปรากฏ.....	27
2.13.3 ตัวประกอบกำลังไฟฟ้า.....	27
2.13.3.1 ตัวประกอบกำลังไฟฟ้าแบบนำหน้า.....	28
2.13.3.2 ตัวประกอบกำลังไฟฟ้าแบบล้าหลัง.....	28

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา V และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	หน้า
บทที่ 3 วิธีการดำเนินงาน.....	29
3.1 การศึกษาเบื้องต้น.....	29
3.1.1 การเก็บรวบรวมข้อมูล.....	29
3.1.2 ฮาร์ดแวร์ที่ใช้ในการดำเนินงาน.....	29
3.1.3 ซอฟต์แวร์ที่ใช้ในการดำเนินงาน.....	30
3.2 ขั้นตอนการดำเนินงาน.....	30
3.2.1 การเก็บรวบรวมข้อมูล.....	30
3.2.2 เชื่อมต่ออุปกรณ์ฮาร์ดแวร์.....	30
3.2.3 ติดตั้งอุปกรณ์วัดเข้ากับเครื่องปรับอากาศ.....	31
3.2.4 ติดตั้งโปรแกรมซอฟต์แวร์.....	32
3.2.5 สร้างฐานข้อมูลบน MySQL.....	32
3.2.6 สร้างไฟล์ PHP เพื่อเพิ่มข้อมูลแบบ GET ลงใน MySQL.....	34
3.2.7 เขียนโปรแกรมให้ NodeMCU ส่งข้อมูลจากโมดูลเซนเซอร์ไปยัง MySQL.....	35
3.2.8 การสร้างเว็บแสดงผล.....	36
บทที่ 4 ผลการทดลอง.....	38
4.1 การทดสอบส่งข้อมูลจากไมโครคอนโทรลเลอร์ไปเก็บบนฐานข้อมูล MySQL.....	38
4.2 การทดสอบความถูกต้องของข้อมูล.....	40
4.2.1 การทดสอบความถูกต้องของการวัดกระแสไฟฟ้า.....	40
4.2.2 การทดสอบความถูกต้องของการวัดแรงดันไฟฟ้า.....	41
4.3 การแสดงผลกราฟข้อมูลบนเว็บเบราว์เซอร์.....	41
บทที่ 5 สรุปผลการทดลองและข้อเสนอแนะ.....	46
5.1 สรุปผลการทดลอง.....	46
5.2 ข้อเสนอแนะ.....	47
บรรณานุกรม.....	48
ภาคผนวก ก การเขียนโปรแกรม.....	49
ภาคผนวก ข Datasheet.....	74

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา VI และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่	หน้า
2.1 แสดงข้อกำหนดของโมดูลเซนเซอร์วัดแรงดัน ZMPT101B.....	7
2.2 แสดงข้อมูลทางด้านเทคนิคของโมดูลเซนเซอร์วัดอุณหภูมิและความชื้น AM2302.....	8
2.3 แสดงชนิดและขนาดของข้อมูลที่ใช้ในภาษา C.....	18
2.4 แสดงรหัสรูปแบบที่ใช้ในฟังก์ชัน printf.....	19
2.5 แสดง Escape sequence ต่าง ๆ ที่มีใช้ในภาษา C.....	19
2.6 แสดงประเภทของตัวดำเนินการทางคณิตศาสตร์.....	20
2.7 แสดงตัวดำเนินการสัมพันธ์และตัวดำเนินการตรรกะ.....	21
2.8 แสดงตัวประกอบดำเนินการ.....	21
4.1 ผลการทดลองเปรียบเทียบความถูกต้องของการวัดกระแสไฟฟ้ากับอุปกรณ์มาตรฐาน.....	40
4.2 ผลการทดลองเปรียบเทียบความถูกต้องของการวัดแรงดันไฟฟ้ากับอุปกรณ์มาตรฐาน.....	41



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา **VI** ต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่	หน้า
2.1 การจัดหาของบอร์ด NodeMCU V3.....	4
2.2 การจัดหาของไอซี MCP3008.....	5
2.3 แสดงบล็อกไดอะแกรมของไอซี ACS712.....	6
2.4 แสดงขาพอร์ตและส่วนประกอบของโมดูลเซนเซอร์วัดกระแสไฟฟ้า ACS712-20A.....	6
2.5 การจัดหาของโมดูลเซนเซอร์วัดแรงดันไฟฟ้า ZMPT101B.....	6
2.6 แสดงการจัดขาโมดูลเซนเซอร์วัดอุณหภูมิและความชื้น AM2302.....	8
2.7 บล็อกไดอะแกรมโมดูลนาฬิกา DS3231.....	9
2.8 แสดงความสัมพันธ์ระหว่างระบบจัดการฐานข้อมูล โปรแกรมต่าง ๆ และฐานข้อมูล.....	10
2.9 แสดงการส่งข้อมูลจากผู้ใช้งานผ่านระบบอินเทอร์เน็ตผ่านเว็บเซิร์ฟเวอร์..... เพื่อเข้าสู่ระบบจัดการฐานข้อมูล MySQL	15
2.10 รูปคลื่นไซน์ของแรงดันไฟฟ้ากระแสสลับ.....	25
2.11 กราฟสุ่มตัวอย่างแรงดันไฟฟ้าและกระแสไฟฟ้าเมื่อทำการวัดอย่างต่อเนื่อง.....	25
2.12 สามเหลี่ยมกำลังไฟฟ้า.....	27
2.13 แสดงลักษณะรูปคลื่นระหว่างกระแสไฟฟ้าและแรงดันไฟฟ้า.....	28
2.14 แสดงสามเหลี่ยมกำลังไฟฟ้า.....	28
3.1 บล็อกไดอะแกรมการเชื่อมต่อโมดูลเซนเซอร์เข้ากับ..... ไมโครคอนโทรลเลอร์ NodeMCU V3	30
3.2 การเชื่อมต่อโมดูลเซนเซอร์ต่าง ๆ เข้ากับไมโครคอนโทรลเลอร์ NodeMCU V3.....	31
3.3 การติดตั้งอุปกรณ์วัดเข้ากับเครื่องปรับอากาศ.....	31
3.4 แสดงการเปิดใช้งาน Apache และ MySQL บนโปรแกรม XAMPP Control Panel.....	32
3.5 แสดงการเรียกใช้ phpMyadmin บนเว็บเบราว์เซอร์.....	33
3.6 แสดงการสร้างฐานข้อมูลบน phpMyAdmin.....	33
3.7 แสดงการสร้างตารางบนฐานข้อมูล MySQL.....	33
3.8 แสดงการสร้างคอลัมน์ที่ต้องการเก็บค่าบน MySQL.....	34
3.9 แสดงคอลัมน์ที่สร้างขึ้นเพื่อรับข้อมูลจากไมโครคอนโทรลเลอร์เก็บไว้ใน MySQL.....	34
3.10 แสดงการเขียนโปรแกรมบน Notepad ด้วย PHP เพื่อติดต่อฐานข้อมูล MySQL.....	34
3.11 ผังงานแสดงการทำงานของไมโครคอนโทรลเลอร์.....	35
3.12 แสดงการเขียนโปรแกรมใน Arduino IDE.....	36
3.13 แสดงข้อมูลที่วัดได้จากโมดูลเซนเซอร์ผ่านหน้าเว็บเซิร์ฟเวอร์ phpMyAdmin.....	36

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา VIII ต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่	หน้า
3.14 โครงสร้างของเว็บแสดงผล.....	37
4.1 ข้อมูลจำเพาะของเครื่องปรับอากาศ SHARP รุ่น Standard Plasma AU-PR13.....	38
4.2 ตารางข้อมูลที่ถูกเก็บบนฐานข้อมูล MySQL ขณะเครื่องปรับอากาศทำงาน.....	39
4.3 ตารางข้อมูลที่ถูกเก็บบนฐานข้อมูล MySQL ขณะเครื่องปรับอากาศตัดการทำงาน.....	39
4.4 แสดงหน้าเว็บเพจเพื่อเลือกช่วงวันที่ที่ต้องการ.....	43
4.5 แสดงกราฟการวัดกระแสไฟฟ้าของเครื่องปรับอากาศในช่วงวันที่ 2 พฤษภาคม 2561.....	43
4.6 แสดงกราฟการวัดแรงดันไฟฟ้าของเครื่องปรับอากาศในช่วงวันที่ 2 พฤษภาคม 2561.....	43
4.7 แสดงกราฟการวัดอุณหภูมิห้องในช่วงวันที่ 2 พฤษภาคม 2561.....	44
4.8 แสดงกราฟค่ากำลังไฟฟ้าในช่วงวันที่ 2 พฤษภาคม 2561.....	44
4.9 ตารางข้อมูลที่ถูกเก็บบนฐานข้อมูล MySQL ขณะเครื่องปรับอากาศทำงาน.....	44
4.10 ตารางข้อมูลที่ถูกเก็บบนฐานข้อมูล MySQL ขณะเครื่องปรับอากาศตัดการทำงาน.....	45
4.11 ตารางข้อมูลที่ถูกเก็บบนฐานข้อมูล MySQL ขณะเกิดกระแสสตาร์ท.....	45
4.12 แสดงกราฟค่าตัวประกอบกำลังไฟฟ้าในช่วงวันที่ 2 พฤษภาคม 2561.....	45

บทที่ 1

บทนำ

1.1 ความสำคัญของปริญญาโท

ข้อมูลเกี่ยวกับการใช้พลังงานของอุปกรณ์ต่าง ๆ นั้น มีประโยชน์อย่างมากในการวิเคราะห์หรือตรวจสอบประสิทธิภาพในการทำงานของอุปกรณ์ต่าง ๆ รวมถึงอาจนำไปใช้ประโยชน์ในการวิเคราะห์ปัญหาหรือความผิดปกติของอุปกรณ์นั้น ๆ ได้อีกด้วย ซึ่งในโครงการนี้เลือกที่จะใช้เครื่องปรับอากาศมาเป็นกรณีศึกษา เนื่องจากเป็นอุปกรณ์ที่มีการใช้งานกันอย่างแพร่หลายทั้งในภาคครัวเรือน ภาคธุรกิจรวมถึงภาคอุตสาหกรรม ซึ่งโดยทั่วไปแล้วหากจะพิจารณาความผิดปกติของเครื่องปรับอากาศนั้นช่างจะใช้การวัดความดันของน้ำยาภายในระบบปรับอากาศมาใช้ในการวิเคราะห์ แต่เนื่องจากปริมาณกระแสไฟฟ้าหรือการใช้พลังงานของเครื่องปรับอากาศนั้น มีความสัมพันธ์โดยตรงกับความดันของน้ำยาภายในระบบปรับอากาศ โครงการนี้จึงมีแนวคิดที่จะวัดปริมาณของกระแสไฟฟ้ายรวมที่เครื่องปรับอากาศใช้งาน โดยจะทำการบันทึกค่ากระแสไฟฟ้าและอุณหภูมิในขณะที่มีการใช้งาน เก็บบันทึกเป็นข้อมูลลงบนฐานข้อมูล ซึ่งสามารถดูย้อนหลังเพื่อวิเคราะห์แนวโน้มของปริมาณการใช้กระแสไฟฟ้าในรูปแบบกราฟหรือตารางเปรียบเทียบในลักษณะต่าง ๆ เพื่อใช้สำหรับการวิเคราะห์ต่อไป

1.2 วัตถุประสงค์ของปริญญาโท

เพื่อสร้างอุปกรณ์การวัดที่สามารถส่งข้อมูลปริมาณกระแสไฟฟ้า แรงดันไฟฟ้า กำลังไฟฟ้า ค่าตัวประกอบกำลังไฟฟ้าและการเปลี่ยนแปลงของอุณหภูมิภายในห้อง ขณะใช้งานเครื่องปรับอากาศ ผ่านการสื่อสารแบบไร้สาย เข้าสู่ระบบอินเทอร์เน็ตเพื่อนำค่าที่ได้ไปเก็บที่เครื่องแม่ข่าย รวมถึงสามารถแสดงผลข้อมูลในรูปแบบกราฟให้กับผู้ใช้งานผ่านทางคอมพิวเตอร์ได้

1.3 ขอบเขตของปริญญาโท

1. สามารถส่งข้อมูลจากไมโครคอนโทรลเลอร์เข้าสู่ฐานข้อมูลบนเครื่องแม่ข่ายผ่านระบบอินเทอร์เน็ตได้
2. สามารถแสดงผลรายงานบนคอมพิวเตอร์ผ่านหน้าเว็บเบราว์เซอร์ได้

1.4 ขั้นตอนการศึกษา

1. ศึกษาการวัดกระแสไฟฟ้า แรงดันไฟฟ้าและอุณหภูมิ
2. ศึกษาการใช้งานไมโครคอนโทรลเลอร์พร้อมเครือข่ายแบบไร้สาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ศึกษาการเขียนโปรแกรมควบคุมไมโครคอนโทรลเลอร์ด้วยภาษาซี
4. ศึกษาการทำให้เครื่องคอมพิวเตอร์เป็นเว็บเซิร์ฟเวอร์
5. ศึกษาการเขียนโปรแกรมบนเครื่องแม่ข่ายด้วยภาษา PHP
6. ศึกษาการเขียนเว็บไซต์ด้วยภาษา HTML, PHP และ JavaScript

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1. โมดูลเซนเซอร์วัดกระแสไฟฟ้า แรงดันไฟฟ้าและอุณหภูมิ
2. การเขียนโปรแกรมบนไมโครคอนโทรลเลอร์พร้อมเครือข่ายแบบไร้สาย
3. การส่งข้อมูลเข้าสู่เครื่องแม่ข่ายผ่านระบบอินเทอร์เน็ต
4. การสร้างเว็บเซิร์ฟเวอร์
5. การเขียนโปรแกรมลงบนเครื่องแม่ข่าย
6. การสร้างเว็บไซต์



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ทฤษฎีและหลักการที่เกี่ยวข้อง

2.1 ไมโครคอนโทรลเลอร์

ไมโครคอนโทรลเลอร์ (Microcontroller) คือ อุปกรณ์อิเล็กทรอนิกส์แบบหนึ่ง ซึ่งบรรจุความสามารถที่คล้ายคลึงกับระบบคอมพิวเตอร์ โดยในไมโครคอนโทรลเลอร์ได้รวมเอาหน่วยประมวลผล หน่วยคำนวณทางคณิตศาสตร์และลอจิก วงจรรับสัญญาณอินพุต วงจรขับสัญญาณเอาต์พุต และหน่วยความจำไว้ด้วยกัน ทำให้สามารถนำไปใช้งานแทนวงจรอิเล็กทรอนิกส์ที่ซับซ้อนได้เป็นอย่างดี

2.1.1 บอร์ด NodeMCU V3

NodeMCU คือ แพลตฟอร์มหนึ่งที่ใช้ช่วยในการสร้างโปรเจกต์ Internet of Things ที่ประกอบไปด้วย ตัวบอร์ด (Development Kit) และ ซอฟต์แวร์บนบอร์ด (Firmware) ที่เป็น Open source สามารถเขียนโปรแกรมด้วยภาษา Lua ได้ แต่เมื่อไม่นานมานี้มีนักพัฒนาที่สามารถทำให้ Arduino IDE ใช้งานร่วมกับ NodeMCU ได้ จึงทำให้สามารถเขียนโปรแกรมโดยใช้ภาษา C/C++ ได้

2.1.2 คุณสมบัติของบอร์ด NodeMCU V3

NodeMCU ใช้ ESP8266 เป็นชิพยูนิตสำหรับประมวลผลโปรแกรมต่าง ๆ มีข้อดีกว่า Arduino ตรงที่ตัวมันมีขนาดเล็กกว่า มีพื้นที่เขียนโปรแกรมลงไปมากกว่า ทำให้สามารถประมวลผลโค้ดโปรแกรมได้อย่างรวดเร็ว เหมาะมากสำหรับงาน Smart Home และ Internet of Things โดยบอร์ด NodeMCU [1][5] มีคุณสมบัติดังนี้

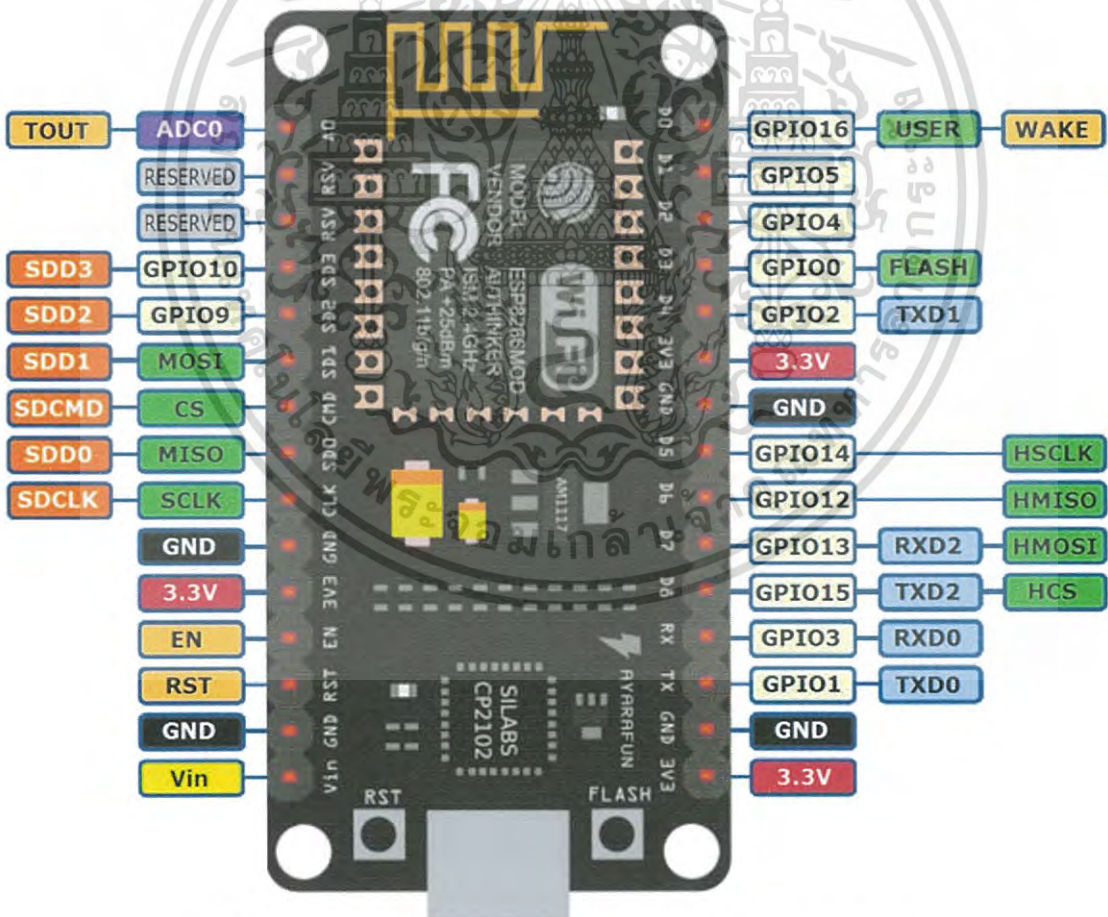
- 1) ใช้ ESP8266-12E เป็นชิพยูนิตสำหรับประมวลผล
- 2) มีพื้นที่หน่วยความจำรวม 4 เมกะไบต์
- 3) ใช้ชิพ USB เบอร์ CH340 ในการติดต่อกับคอมพิวเตอร์เพื่อลงโปรแกรม
- 4) สามารถใช้ไฟเลี้ยงบอร์ดจากพอร์ต USB 5 VDC หรือ จากแหล่งจ่ายไฟภายนอก 5-10 VDC
- 5) ปริมาณการกินกระแส 10 μ A ถึง 170 mA
- 6) ความจำแฟลชสูงสุด 16 เมกะไบต์
- 7) ใช้ระบบปฏิบัติการ Tensilica L106 32 บิต
- 8) ความเร็วระบบปฏิบัติการ 80-160 MHz
- 9) สามารถพัฒนาโปรแกรมโดยใช้โปรแกรม Arduino IDE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.3 ข้อมูลเชิงเทคนิคของบอร์ด NodeMCU V3

NodeMCU V3 ปรับปรุงมาจาก NodeMCU เวอร์ชันเดิม ซึ่งมีขา GPIO, PWM, I2C, 1-Wire, ADC และ SPI เพิ่มเข้ามา การจัดขาของบอร์ด NodeMCU V3 ดังรูปที่ 2.1 ซึ่งมีรายละเอียดดังต่อไปนี้

- 1) ขา GPIO ทั้งหมด 17 ช่อง
- 2) ระดับสัญญาณลอจิกของสัญญาณอินพุตและเอาต์พุต (GPIO) 3.3 โวลต์
- 3) ขา PWM รับส่งสัญญาณดิจิทัล 9 ขา
- 4) ขา ADC 1 ช่อง ความละเอียดของค่าที่อ่านได้ 10 บิต
- 5) มีขา D1 และ D2 สำหรับเชื่อมต่อแบบ I2C
- 6) มีขา D5, D6, D7 และ D8 สำหรับเชื่อมต่อแบบ SPI
- 7) มีปุ่มกด RST เพื่อรีเซ็ตการทำงาน และ Flash สำหรับโปรแกรมเฟิร์มแวร์ใหม่
- 8) ขนาดของบอร์ด 25.4 มิลลิเมตร x 48.26 มิลลิเมตร

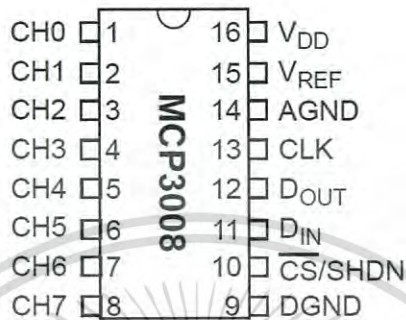


รูปที่ 2.1 การจัดขาของบอร์ด NodeMCU V3 (Smartyshirt, 2018)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.4 ไอซีเบอร์ MCP3008

ไอซีเบอร์ MCP3008 เป็นชิปที่แปลงสัญญาณอนาล็อกเป็นดิจิทัล 8 ช่องสัญญาณ ขนาด 10 บิต สื่อสารด้วย SPI Interface โดยขาของไอซี MCP3008 แสดงตามรูปที่ 2.2 ซึ่งสามารถนำไปประยุกต์ใช้ในการขยายขาสัญญาณอนาล็อกอุปกรณ์ Raspberry Pi หรือ NodeMCU [6]



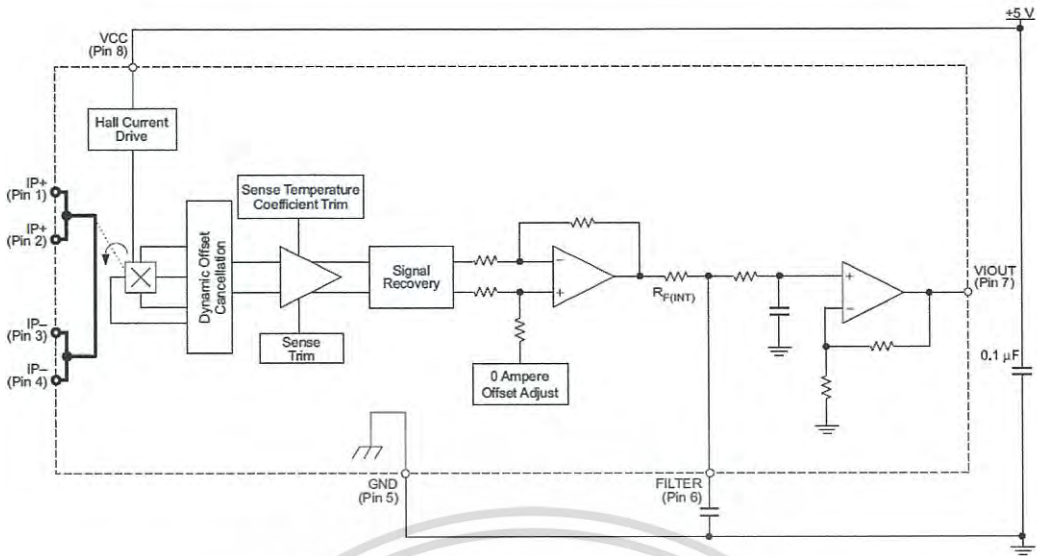
รูปที่ 2.2 การจัดขาของไอซี MCP3008 (Tony DiCola, 2016)

2.2 โมดูลเซนเซอร์วัดกระแสไฟฟ้า ACS712-20A

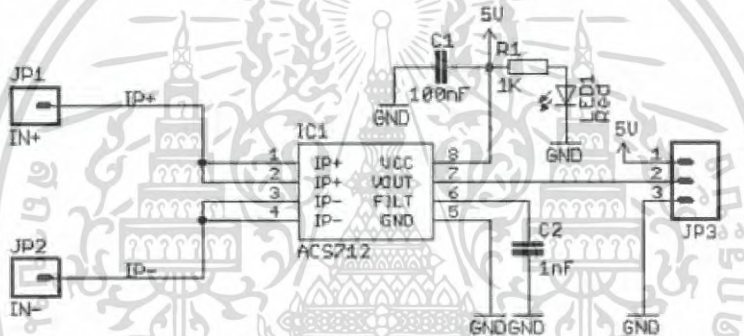
โมดูลเซนเซอร์วัดกระแสไฟฟ้าประกอบด้วยอุปกรณ์หลักที่ใช้วัดกระแสคือไอซี ACS712 ซึ่งใช้หลักการของ Hall Effect เพื่อวัดสนามแม่เหล็กที่เกิดขึ้นจากกระแสไฟฟ้าที่ไหลผ่านอยู่ในรูปแบบของแรงดันไฟฟ้า ซึ่งแรงดันไฟฟ้าที่วัดได้นั้นจะมีค่าน้อยมากจึงต้องมีการขยายสัญญาณรวมอยู่ในไอซีตัวนี้ด้วย ดังรูปที่ 2.3 เพื่อให้ได้สัญญาณแรงดันเอาต์พุตมีขนาดที่สูงขึ้นเมื่อเทียบกับกระแสอินพุต โดยจะมีค่าความไวอยู่ที่ 100 mV/A [7] ในการนำโมดูลเซนเซอร์วัดกระแสไฟฟ้า ACS712-20A ไปใช้งาน จะมีขาต่าง ๆ และส่วนประกอบอื่น ดังรูปที่ 2.4 โดยโมดูลเซนเซอร์วัดกระแสไฟฟ้า ACS712-20A สามารถวัดค่ากระแสไฟฟ้าได้สูงสุด 20 A ซึ่งมีคุณสมบัติอื่น ๆ ดังนี้

- 1) มีสัญญาณรบกวนตามทางเดินต่ำ
- 2) Rise time 5 ไมโครวินาที สำหรับอินพุตกระแสแบบ step
- 3) แบนด์วิดธ์ 80 กิโลเฮิร์ตซ์
- 4) มีความผิดพลาดของเอาต์พุต 1.5% ที่ อุณหภูมิ 25 °C
- 5) อุปกรณ์มีขนาดเล็ก 31 มิลลิเมตร × 13 มิลลิเมตร
- 6) ความต้านทานภายใน 1.2 มิลลิโอห์ม
- 7) ทำงานที่แรงดัน 5 โวลต์
- 8) ความไว 66-185 mV/A ขึ้นอยู่กับอุณหภูมิสภาพแวดล้อม
- 9) แรงดันเอาต์พุตเป็นสัดส่วนโดยตรงกับกระแสไฟฟ้า
- 10) ฮีสเตอร์ซิสที่เกิดจากแม่เหล็กน้อยมากเกือบมีค่าเป็นศูนย์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.3 แสดงบล็อกโตะแกรมของไอซี ACS712 (Allegro MicroSystems, 2017)



รูปที่ 2.4 แสดงขาพอร์ตและส่วนประกอบของโมดูลเซนเซอร์วัดกระแสไฟฟ้า ACS712-20A (Allegro MicroSystems, 2017)

2.3 โมดูลเซนเซอร์วัดแรงดันไฟฟ้า ZMPT101B

โมดูลวัดแรงดันไฟฟ้า ZMPT101B เป็นชุดโมดูลเซนเซอร์วัดแรงดันไฟฟ้ากระแสสลับแบบ Single Phase สูงสุดที่ 250 VAC สัญญาณที่ออกจากโมดูลเป็นสัญญาณอนาล็อก สามารถนำไปต่อเข้ากับขา Analog to Digital Converter ของไมโครคอนโทรลเลอร์ที่ใช้แรงดันอ้างอิง 5 โวลต์ มีวงจรขยายสัญญาณ สามารถปรับขนาดแอมพลิจูดของสัญญาณเอาต์พุตได้ จากการปรับตัวต้านทานปรับค่าได้บนบอร์ดโดยมีข้อกำหนดต่าง ๆ แสดงดังตารางที่ 2.1 และการจัดขาของโมดูล ดังรูปที่ 2.5



รูปที่ 2.5 การจัดขาของโมดูลเซนเซอร์วัดแรงดันไฟฟ้า ZMPT101B (Yanuar M, 2017)

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์โดยผู้ประพันธ์เนื้อหา กรุณาแจ้งเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.1 แสดงข้อกำหนดของโมดูลเซนเซอร์วัดแรงดัน ZMPT101B (Nanjing Zeming, 2015)

ข้อกำหนดทางไฟฟ้า	
Primary Current	2 mA
Secondary Current	2 mA
อัตราส่วนรอบ	1000:1000
ความคลาดเคลื่อนมุมเฟส	$\leq 20^\circ$
ช่วงของกระแส	0 ~ 3 mA
Linearity	0.1%
Accuracy Class	0.2
Rated Burden	$\leq 200 \Omega$
ช่วงความถี่	50 ~ 60 Hz
Dielectric Level	3000VAC/min
DC Resistance at 20 °C	110 Ω
ข้อกำหนดเกี่ยวกับสภาพแวดล้อม	
อุณหภูมิในการเก็บรักษา	-40 °C ~ +130 °C
ฉนวนกันความร้อน	> 100 M Ω

2.4 โมดูลเซนเซอร์วัดอุณหภูมิและความชื้น AM2302

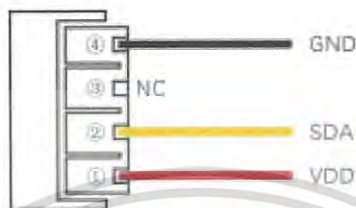
อุปกรณ์เซนเซอร์สำหรับวัดอุณหภูมิและความชื้นสัมพัทธ์ (Temperature & Relative Humidity Sensor) เป็นอุปกรณ์ที่สามารถนำมาประยุกต์ใช้งานทางด้านระบบสมองกลฝังตัวได้หลากหลาย เช่น การวัดและควบคุมอุณหภูมิและความชื้น ระบบบันทึกข้อมูลเกี่ยวกับอุณหภูมิและความชื้นในห้อง เป็นต้น [8] โดยข้อมูลทางเทคนิคแสดงดังตารางที่ 2.2 โมดูลเซนเซอร์วัดอุณหภูมิและความชื้น AM2302 มีคุณสมบัติดังนี้

- 1) ให้เอาต์พุตแบบดิจิตอล
- 2) มีการชดเชยอุณหภูมิตัวในตัวเองทำให้มีค่าความแม่นยำสูง
- 3) มีความโดดเด่นเรื่องเสถียรภาพในการใช้งานที่ยาวนาน
- 4) ไม่ต้องต่ออุปกรณ์อื่น ๆ เพิ่ม เช่น ตัวต้านทานหรือวงจรถยาย
- 5) สามารถลากสายสัญญาณได้ไกลถึงกว่า 100 เมตร
- 6) อัตราการใช้พลังงานต่ำเพราะข้อมูลเป็นแบบดิจิตอล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขาของโมดูลเซนเซอร์วัดอุณหภูมิและความชื้น AM2302 มีทั้งหมด 4 ขา แต่ต่อใช้งานเพียง 3 ขา แสดงดังรูปที่ 2.6 ซึ่งมีรายละเอียดดังต่อไปนี้

- 1) สายไฟสีแดงคือ VDD จ่ายไฟ 3.3-6 โวลต์ เข้าไปเลี้ยงวงจรให้ทำงานได้
- 2) สายไฟสีดำคือ GND ต่อกับกราวด์
- 3) สายไฟสีเหลืองคือสัญญาณข้อมูลโดยที่จะมีสัญญาณเป็นดิจิตอล 8 บิต



รูปที่ 2.6 แสดงการจัดขาโมดูลเซนเซอร์วัดอุณหภูมิและความชื้น AM2302
(Aosong (Guangzhou) Electronics, 2018)

ตารางที่ 2.2 แสดงข้อมูลทางด้านเทคนิคของโมดูลเซนเซอร์วัดอุณหภูมิและความชื้น AM2302
(Thomas Liu, 2018)

รุ่น	AM2302	
แหล่งจ่ายพลังงาน	3.3-6 VDC	
สัญญาณเอาต์พุต	สัญญาณดิจิตอลส่งผ่านสายไฟ 1 เส้น	
อุปกรณ์รับรู้	ตัวเก็บประจุความชื้นแบบโพลีเมอร์	
ย่านการทำงาน	ความชื้น 0-100% RH	อุณหภูมิ -40~80 °C
ความแม่นยำ	ความชื้น $\pm 2\%$ RH (สูงสุด $\pm 5\%$ RH)	อุณหภูมิ ± 0.5 °C
ค่าความละเอียดหรือความไว	ความชื้น 0.1% RH	อุณหภูมิ 0.1 °C
ความสามารถในการทวนซ้ำ	ความชื้น $\pm 1\%$ RH	อุณหภูมิ ± 0.2 °C
ฮิสเตอร์รีซิสของความชื้น	$\pm 0.3\%$ RH	
ความเสถียรในระยะยาว	$\pm 0.5\%$ RH/ปี	

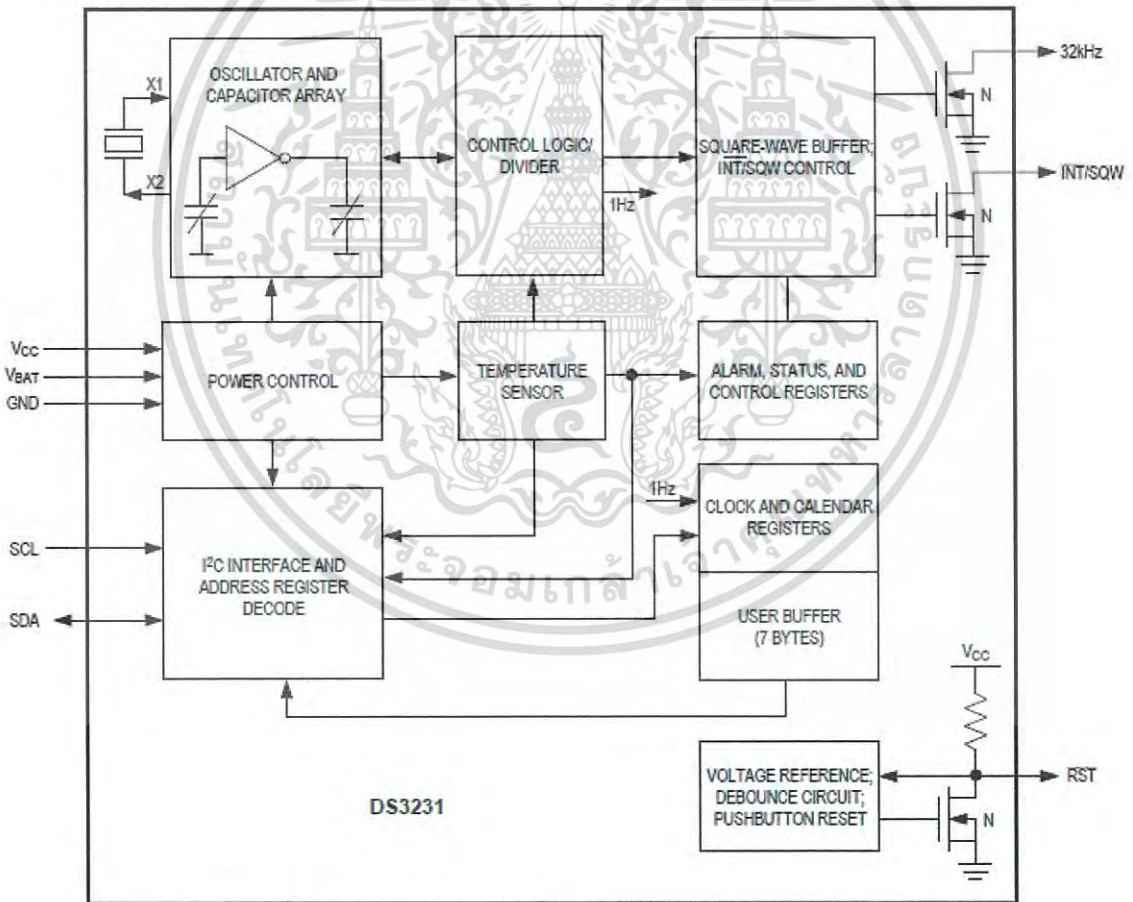
2.5 โมดูลนาฬิกา DS3231

DS3231 เป็นโมดูลนาฬิกาแบบเวลาจริง (Real Time Clock: RTC) ที่มีความถูกต้องแม่นยำสูง เพราะข้างในมีวงจรวัดอุณหภูมิ เพื่อนำอุณหภูมิจากสภาพแวดล้อมมาคำนวณชดเชยความถี่ของคริสตัล ที่ถูกรบกวนจากอุณหภูมิภายนอก มาพร้อมแบตเตอรี่ ใช้งานได้แม้ไม่มีแหล่งจ่ายไฟจากภายนอก สามารถตั้งค่า วัน เวลา ได้อย่างง่าย นอกจากจะแสดงวันและเวลาได้อย่างแม่นยำแล้ว โมดูลนี้ยังสามารถ แสดงอุณหภูมิภายนอกได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การนำโมดูลนาฬิกา DS3231 ไปใช้งานต้องต่อเข้ากับขา Vcc, GND, SCL และ SDA ของโมดูลตามบล็อกไดอะแกรม รูปที่ 2.7 [9] นอกจากนี้ยังมีรายละเอียด ดังต่อไปนี้

- 1) ใช้แรงดันไฟเลี้ยง 3.3 โวลต์
- 2) มีแบตเตอรี่สำรอง
- 3) ใช้พลังงานต่ำ
- 4) เชื่อมต่อแบบบัส I²C และมีความเร็วถึง 400 KHz
- 5) ภายในมีวงจรสร้างสัญญาณนาฬิกา ความถี่ 32 KHz
- 6) มีความแม่นยำ (Accuracy) ± 2 ppm ในช่วงอุณหภูมิ 0 °C ถึง 40 °C และ ± 3.5 ppm สำหรับอุณหภูมิ -40 °C ถึง 85 °C
- 7) สามารถวัดค่าอุณหภูมิได้ข้อมูลดิจิทัลแบบ 10 บิต ความละเอียด 0.25 °C แต่มีความแม่นยำ ± 3 °C

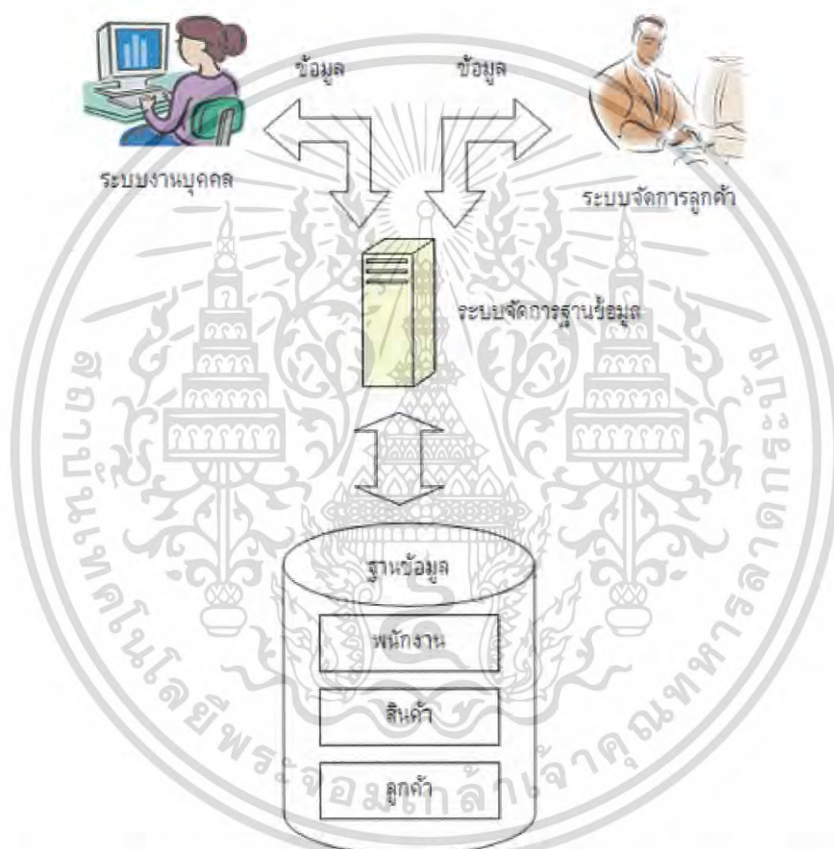


รูปที่ 2.7 บล็อกไดอะแกรมโมดูลนาฬิกา DS3231 (Maxim Integrated, 2015)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6 ระบบจัดการฐานข้อมูล

ระบบจัดการฐานข้อมูล (Database Management System: DBMS) หมายถึง ซอฟต์แวร์ของระบบที่ใช้ในการควบคุม ดูแลและจัดการฐานข้อมูล โดยที่โปรแกรมของระบบงานต่าง ๆ ที่จะเข้ามาใช้ข้อมูลในฐานข้อมูลจะต้องผ่านซอฟต์แวร์ที่ดูแลฐานข้อมูล (ระบบจัดการฐานข้อมูล) แสดงดังรูปที่ 2.8 เพื่อความปลอดภัยของข้อมูลและความถูกต้องของข้อมูล อีกทั้งซอฟต์แวร์ที่ดูแลฐานข้อมูล ยังมีหน้าที่ในการเปลี่ยนแปลงรูปแบบตามความต้องการของผู้ใช้ ให้อยู่ในรูปแบบที่สามารถทำงานได้กับระบบฐานข้อมูลตามความประสงค์ของโปรแกรมหรือผู้ใช้แต่ละคน [10]



รูปที่ 2.8 แสดงความสัมพันธ์ระหว่างระบบจัดการฐานข้อมูล โปรแกรมต่าง ๆ และฐานข้อมูล
(สุกิจ คูชัยสิทธิ์, 2561)

2.6.1 หน้าที่ของระบบจัดการฐานข้อมูล

- 1) ช่วยกำหนดและเก็บโครงสร้างฐานข้อมูล (Define and Store Database Structure) ระบบจัดการฐานข้อมูลจะช่วยกำหนดและเก็บโครงสร้างฐานข้อมูล เพื่อให้ข้อมูลถูกจัดเก็บตามโครงสร้างของฐานข้อมูล และทำให้เกิดความถูกต้องตามกฎเกณฑ์ของระบบ โดยโครงสร้างของฐานข้อมูลนั้น ๆ จะถูกออกแบบโดยผู้ดูแลระบบฐานข้อมูลกับผู้วิเคราะห์และออกแบบระบบ

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์หรือการแจ้งงานเพื่อการศึกษาเท่านั้น เมื่ออนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 2) การบรรจุข้อมูลจากฐานข้อมูล (Load Database) เมื่อมีการเพิ่มข้อมูลหรือมีการประมวลผลที่เกิดจากการทำงานของเหล่าโปรแกรมประยุกต์ต่าง ๆ ระบบจัดการฐานข้อมูลจะทำการรับและเก็บข้อมูลไว้ในฐานข้อมูล เพื่อใช้ในการประมวลผลครั้งต่อไป
- 3) เก็บและดูแลข้อมูล (Store and Maintain Data) เมื่อมีการทำงาน ข้อมูลในระบบฐานข้อมูลจะถูกจัดเก็บและรวบรวมไว้ด้วยกัน โดยมีระบบจัดการฐานข้อมูลเป็นผู้ดูแลรักษาข้อมูล
- 4) ประสานงานกับระบบปฏิบัติการ (Operating Systems) ระบบปฏิบัติการเป็นโปรแกรมระบบของเครื่อง ที่คอยควบคุมการทำงานของอุปกรณ์คอมพิวเตอร์หรือโปรแกรมต่าง ๆ ส่วนระบบจัดการฐานข้อมูลจะทำการประสานงานกับระบบปฏิบัติการในการเรียกใช้ แก้ไขข้อมูล หรือออกรายงานที่ต้องการตามผู้ใช้
- 5) ช่วยควบคุมความปลอดภัย (Security Control) ในระบบจัดการฐานข้อมูล จะมีวิธีควบคุมการเรียกใช้หรือแก้ไขเปลี่ยนแปลงข้อมูลของผู้ใช้ในระบบ ซึ่งสามารถเรียกใช้ข้อมูลมาแก้ไขได้แตกต่างกัน เพื่อเป็นการป้องกันความเสียหายที่อาจจะเกิดขึ้นกับฐานข้อมูล
- 6) การจัดทำข้อมูลสำรองและการกู้ (Backup and Recovery) ระบบจัดการฐานข้อมูลจะมีโปรแกรมเพื่อสนับสนุนการสำรองและกู้คืนของข้อมูล เพื่อให้แน่ใจในความปลอดภัยและความมั่นคงของข้อมูลในระบบ ระบบจัดการฐานข้อมูลจะทำการกู้ข้อมูลในฐานข้อมูลคืนมา หลังจากระบบเกิดความล้มเหลว (Failure) เช่น กระแสไฟฟ้าเกิดขัดข้อง เป็นต้น
- 7) ควบคุมการใช้ข้อมูลพร้อมกันของผู้ใช้ในระบบ (Concurrency Control) ในระบบคอมพิวเตอร์ที่ผู้ใช้หลายคนสามารถเรียกใช้ข้อมูลพร้อมกัน ระบบจัดการฐานข้อมูลที่มีคุณสมบัตินี้ จะทำการควบคุมการใช้ข้อมูลพร้อมกันของผู้ใช้หลายคนในเวลาเดียวกันได้ โดยมีการควบคุมอย่างถูกต้องเหมาะสม เช่น ถ้ามีการแก้ไขข้อมูลนั้นยังไม่เรียบร้อย ผู้ใช้อื่น ๆ ที่ต้องการเรียกใช้ข้อมูลนี้ จะต้องรอจนกว่าการแก้ไขเสร็จเรียบร้อย เพื่อป้องกันไม่ให้เกิดปัญหาการเรียกใช้ข้อมูลที่ไม่ถูกต้อง
- 8) ควบคุมความบูรณาภาพของข้อมูล (Integrity Control) ระบบจัดการฐานข้อมูลจะสนับสนุนและควบคุมความถูกต้องของข้อมูล ตั้งแต่การลดความซ้ำซ้อนของข้อมูลไปจนถึงความไม่สอดคล้องกันของข้อมูล ความสัมพันธ์ของข้อมูลที่เก็บในพจนานุกรมข้อมูลจะถูกนำมาใช้ในการควบคุมความถูกต้องของข้อมูลด้วย
- 9) ทำหน้าที่จัดทำพจนานุกรมข้อมูล (Data Dictionary) ระบบจัดการฐานข้อมูลจะทำการจัดเก็บนิยามของข้อมูลและความสัมพันธ์ระหว่างข้อมูล ไว้ในพจนานุกรม

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อการศึกษาเท่านั้น เมื่อผู้ผู้ใดเห็นไปใช้ประโยชน์ในการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูลเป็นสารสนเทศที่บอกเกี่ยวกับโครงสร้างของฐานข้อมูล โปรแกรมประยุกต์ ทั้งหมดที่ต้องการเข้าถึงข้อมูล ในฐานข้อมูลจะต้องทำงานผ่านระบบจัดการ ฐานข้อมูลโดยที่ระบบจัดการฐานข้อมูลจะใช้พจนานุกรมข้อมูล เพื่อค้นหา โครงสร้าง ตลอดจนส่วนประกอบของข้อมูลและความสัมพันธ์ที่ต้องการ นอกจากนั้นแล้วการเปลี่ยนแปลงใด ๆ ที่มีต่อโครงสร้างฐานข้อมูลจะถูกบันทึก ไว้โดยอัตโนมัติในพจนานุกรมข้อมูล ทำให้ไม่ต้องไปเปลี่ยนแปลง แกไขโปรแกรม เมื่อโครงสร้างข้อมูลมีการเปลี่ยนแปลง

2.6.2 องค์ประกอบของระบบฐานข้อมูล

ระบบฐานข้อมูล เป็นระบบที่ต้องนำเอาคอมพิวเตอร์เข้ามาช่วยในการจัดเก็บข้อมูล โดยมีซอฟต์แวร์หรือโปรแกรมช่วยจัดการข้อมูลเหล่านี้ เพื่อให้ได้ข้อมูลอย่างรวดเร็วตามที่ผู้ต้องการ องค์ประกอบของฐานข้อมูลแบ่งออกเป็น 5 ประเภท คือ

2.6.2.1 ฮาร์ดแวร์ (Hardware)

ระบบฐานข้อมูลที่มีประสิทธิภาพสูงควรมีฮาร์ดแวร์ต่าง ๆ ที่มีประสิทธิภาพ พร้อมจะอำนวยความสะดวกในการจัดการระบบฐานข้อมูลได้อย่างมีประสิทธิภาพ ตั้งแต่ความเร็วของ หน่วยประมวลผลกลาง ขนาดของหน่วยความจำหลัก อุปกรณ์นำข้อมูลเข้า และออกรายงาน รวมถึง หน่วยความจำสำรองที่จะรองรับการประมวลผลข้อมูลในระบบได้อย่างมีประสิทธิภาพ

2.6.2.2 โปรแกรม (Program)

การประมวลผลฐานข้อมูลจะใช้โปรแกรมที่แตกต่างกัน ทั้งนี้ขึ้นอยู่กับระบบ คอมพิวเตอร์ที่ใช้โปรแกรมที่ทำหน้าที่ควบคุมดูแลการสร้าง การเรียกใช้ข้อมูล การจัดการทำรายงาน การปรับเปลี่ยนแก้ไข โครงสร้างการควบคุมความปลอดภัยของข้อมูลก็คือระบบจัดการฐานข้อมูล (Database Management Systems: DBMS) ซึ่งเป็นโปรแกรมที่ทำหน้าที่เป็นสื่อกลางระหว่างผู้ใช้ กับโปรแกรมประยุกต์ต่าง ๆ ที่ใช้ระบบฐานข้อมูล

2.6.2.3 ข้อมูล (Data)

ข้อมูลจะถูกรวบรวมไว้ในระบบฐานข้อมูล ซึ่งระบบฐานข้อมูลจะเป็น ศูนย์กลางของข้อมูลอย่างมีระบบและสามารถใช้ข้อมูลนี้ร่วมกันได้ โดยผู้ใช้ข้อมูลในระบบฐานข้อมูล จะมองภาพข้อมูลในลักษณะที่แตกต่างกันไป

2.6.2.4 บุคลากร (People)

บุคลากรที่เกี่ยวข้องกับฐานข้อมูล ได้แก่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 1) ผู้ใช้ทั่วไป (User) เป็นผู้ใช้ระบบฐานข้อมูล โดยผ่านทางโปรแกรมประยุกต์ ที่นักเขียนโปรแกรมประยุกต์ได้จัดเตรียมไว้ให้เรียบร้อยแล้ว เช่น พนักงานที่เคาน์เตอร์ เป็นต้น
- 2) พนักงานปฏิบัติการ (Operator) เป็นผู้ป้อนข้อมูลเข้าสู่ระบบฐานข้อมูลผ่านโปรแกรมประยุกต์
- 3) นักวิเคราะห์และออกแบบระบบ (System Analyst) เป็นผู้ที่ทำหน้าที่วิเคราะห์และออกแบบระบบฐานข้อมูล และระบบงานที่จะนำระบบฐานข้อมูลมาใช้
- 4) ผู้เขียนโปรแกรม (Programmer) เป็นผู้เขียนโปรแกรมประยุกต์ที่ใช้งานต่าง ๆ เพื่อให้การจัดเก็บ การเรียกใช้ข้อมูลเป็นไปตามความต้องการของผู้ใช้
- 5) ผู้บริหารฐานข้อมูล (Database Administrator: DBA) เป็นผู้ที่ทำหน้าที่บริหารและควบคุมการบริหารงานของระบบฐานข้อมูลทั้งหมด เป็นผู้ที่จะต้องตัดสินใจว่าจะรวบรวมข้อมูลอะไรเข้าในระบบ จัดเก็บโดยวิธีใด เทคนิคการเรียกใช้ข้อมูล กำหนดระบบการรักษาความปลอดภัยของข้อมูล กำหนดสิทธิในการเข้าใช้ระบบฐานข้อมูลให้กับผู้ใช้ กำหนดกฎข้อบังคับในการบูรณาการข้อมูล การสร้างระบบข้อมูลสำรอง การกู้ และประสานงานกับผู้ใช้ว่ามีความต้องการใช้ข้อมูลอย่างไร รวมถึงนักวิเคราะห์และออกแบบระบบและโปรแกรมเมอร์ประยุกต์ใช้งาน เพื่อให้การบริหารระบบฐานข้อมูลเป็นไปอย่างมีประสิทธิภาพ

2.6.2.5 ขั้นตอนปฏิบัติงาน (Procedures)

ระบบฐานข้อมูลควรจะมีการจัดทำเอกสาร ที่ระบุขั้นตอนการทำงานของหน้าทำงานต่าง ๆ ในระบบฐานข้อมูล ข้อมูลทั้งในสภาวะปกติและในสภาวะที่ระบบเกิดปัญหา ซึ่งเป็นขั้นตอนการปฏิบัติงานสำหรับบุคลากรในทุกระดับขององค์กร เมื่อเกิดปัญหาก็สามารถนำเอกสารที่บันทึกการทำงานมาตรวจสอบได้ เพื่อการแก้ไขข้อผิดพลาดได้อย่างรวดเร็วและทำการบันทึกวิธีการแก้ไขเมื่ออาจจะเกิดข้อพลาดนั้นขึ้นซ้ำ

2.6.3 ข้อดีในการใช้ระบบฐานข้อมูล

การมีระบบฐานข้อมูลมาใช้งานนั้นจะมีข้อดี ดังนี้

- 1) หลีกเลี่ยงความขัดแย้งของข้อมูล การจัดเก็บข้อมูลแบบฐานข้อมูลจะมีการจัดเก็บไว้ที่เดียว ถ้ามีการแก้ไขข้อมูลก็จะแก้ที่เดียว ทำให้โอกาสที่ข้อมูลขัดแย้งกันไม่มี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 2) สามารถใช้ข้อมูลร่วมกันได้ การจัดเก็บข้อมูลแบบฐานข้อมูลจะมีการจัดเก็บไว้ที่เดียว ทำให้การเรียกใช้ข้อมูลจากโปรแกรมประยุกต์ต่าง ๆ หลายที่สามารถเรียกใช้ข้อมูลที่เดียวและร่วมกันได้ จะทำให้ได้ข้อมูลที่ทันสมัย
- 3) สามารถลดความซ้ำซ้อนของข้อมูล การจัดเก็บข้อมูลแบบฐานข้อมูลจะมีการจัดเก็บไว้ที่เดียว ถ้ามีการแก้ไขข้อมูลก็จะแก้ที่เดียวทำให้ลดปัญหาเรื่องความซ้ำซ้อนโดยที่มีระบบจัดการฐานข้อมูล ช่วยควบคุมความซ้ำซ้อนในการจัดเก็บและการประมวลผลทำให้ข้อมูลมีความน่าเชื่อถือ
- 4) การรักษาความถูกต้องเชื่อถือได้ของข้อมูล ซึ่งในระบบจัดการข้อมูลสามารถระบุกฎเกณฑ์ เพื่อควบคุมความผิดพลาดที่อาจจะเกิดขึ้นได้ ทำให้เกิดความน่าเชื่อถือข้อมูล
- 5) สามารถกำหนดความเป็นมาตรฐานเดียวกัน การเก็บข้อมูลไว้ที่เดียวกันทำให้สามารถกำหนดและ ควบคุมความมีมาตรฐานของข้อมูลให้เป็นไปในลักษณะเดียวกันได้ เพราะในระบบฐานข้อมูลจะมีกลุ่มบุคคลที่คอยบริหารระบบฐานข้อมูล กำหนดมาตรฐานต่าง ๆ ในการจัดเก็บข้อมูลไปในลักษณะเดียวกัน เช่น โครงสร้างข้อมูล ชนิดของข้อมูลที่จัดเก็บ เป็นต้น
- 6) สามารถกำหนดระบบความปลอดภัยของข้อมูลได้ ผู้บริหารระบบฐานข้อมูลสามารถกำหนดระดับการเรียกใช้ข้อมูลของผู้ใช้แต่ละคน ให้แตกต่างกันตามหน้าที่ความรับผิดชอบได้ง่าย และข้อมูลสำคัญ ๆ หรือเป็นความลับก็จะให้สิทธิเฉพาะผู้ใช้บางคนได้เห็นได้ใช้เท่านั้น
- 7) ความเป็นอิสระของข้อมูลและโปรแกรม

2.6.4 ข้อเสียในการใช้ระบบฐานข้อมูล

การมีระบบฐานข้อมูลมาใช้งานนั้นจะมีข้อเสีย ดังนี้

- 1) ระบบฐานข้อมูลมีต้นทุนสูง ระบบฐานข้อมูลก่อให้เกิดต้นทุนสูงขึ้น เช่น ค่าใช้จ่ายซอฟต์แวร์ที่ใช้ในการจัดการระบบฐานข้อมูล ค่าใช้จ่ายบุคลากรในการปฏิบัติการ ค่าใช้จ่ายในการบำรุงดูแลระบบฐานข้อมูลและค่าใช้จ่ายฮาร์ดแวร์ เป็นต้น
- 2) ระบบฐานข้อมูลมีความซับซ้อน ระบบจัดการฐานข้อมูลช่วยอำนวยความสะดวกในการจัดการข้อมูลได้ดี ซึ่งกระบวนการนั้นทำให้เกิดความสลับซับซ้อนสูงได้ เช่นการจัดเก็บข้อมูล การออกแบบฐานข้อมูล การเขียนโปรแกรม เป็นต้น
- 3) การเสี่ยงต่อการหยุดชะงักของระบบ เนื่องจากข้อมูลอาจถูกเก็บไว้ในลักษณะเป็นศูนย์รวม (Centralized Database System) หากระบบเกิดความล้มเหลว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บางส่วนทำให้ในระบบฐานข้อมูลเกิดการหยุดชะงัก มีผลกระทบต่อการทำงานของหยุดชะงักตามไปด้วย

- 4) การกู้คืนทำได้ยาก ต้องใช้เครื่องมือต่าง ๆ มากขึ้น

2.6.5 โปรแกรมระบบจัดการฐานข้อมูล MySQL

MySQL คือ โปรแกรมระบบจัดการฐานข้อมูล ที่พัฒนาโดยบริษัท MySQL AB มีหน้าที่เก็บข้อมูลอย่างเป็นระบบ รองรับคำสั่ง SQL เป็นเครื่องมือสำหรับเก็บข้อมูล ที่ต้องใช้ร่วมกับเครื่องมือหรือโปรแกรมอื่นอย่างบูรณาการ แสดงดังรูปที่ 2.9 เพื่อให้ได้ระบบงานที่รองรับความต้องการของผู้ใช้ เช่นทำงานร่วมกับเครื่องบริการเว็บ เพื่อให้บริการแก่ภาษาสคริปต์ที่ทำงานฝั่งเครื่องบริการ (Server-Side Script) เช่น ภาษา PHP ภาษา asp.net หรือภาษา JSP เป็นต้น หรือทำงานร่วมกับโปรแกรมประยุกต์ เช่น ภาษา Visual basic ภาษา Java หรือภาษา C# เป็นต้น [3] โปรแกรมถูกออกแบบให้สามารถทำงานได้บนระบบปฏิบัติการที่หลากหลายและเป็นระบบฐานข้อมูลแบบ Open Source ที่ถูกนำไปใช้งานมากที่สุด



รูปที่ 2.9 แสดงการส่งข้อมูลจากผู้ใช้งานผ่านระบบอินเทอร์เน็ต

ผ่านเว็บเซิร์ฟเวอร์เพื่อเข้าสู่ระบบจัดการฐานข้อมูล MySQL (Mitchell Anicas, 2014)

2.6.6 โปรแกรม phpMyAdmin

phpMyAdmin คือโปรแกรมที่ถูกพัฒนาโดยใช้ภาษา PHP เพื่อใช้ในการบริหารจัดการฐานข้อมูล MySQL แทนการคีย์คำสั่ง เป็นเครื่องมือในการจัดการฐานข้อมูล MySQL เพื่อให้สามารถจัดการระบบจัดการฐานข้อมูล ที่เป็น MySQL ได้ง่ายและสะดวกยิ่งขึ้น โดย phpMyAdmin ถือเป็นเครื่องมือชนิดหนึ่งในการจัดการนั่นเอง โดยสามารถที่จะทำการสร้างฐานข้อมูลใหม่ หรือทำการสร้างตารางและยังมีฟังก์ชันที่ใช้สำหรับการทดสอบการ Query ข้อมูลด้วยภาษา SQL พร้อมกันนั้นยังสามารถทำการ แทรก (insert), ลบ (delete), อัปเดต (update) หรือแม้กระทั่งใช้คำสั่งต่าง ๆ เหมือนกันกับการใช้ภาษา SQL ในการสร้างตารางข้อมูล

phpMyAdmin เป็นโปรแกรมประเภท MySQL Client ตัวหนึ่งที่ใช้ในการจัดการข้อมูล MySQL ผ่านเว็บเบราว์เซอร์ได้โดยตรง phpMyAdmin ตัวนี้จะทำงานบนเว็บเบราว์เซอร์เป็น PHP Application ที่ใช้ควบคุมจัดการ MySQL Server ความสามารถของ phpMyAdmin คือ

- 1) สร้างและลบ Database
- 2) สร้างและจัดการ Table เช่น แทรก (insert), ลบ (delete), อัปเดต (update)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 3) โหลดเท็กซ์ไฟล์เข้าไปเก็บเป็นข้อมูลในตารางได้
- 4) ทาผลสรุป (Query) ด้วยคำสั่ง SQL

2.7 เว็บเซิร์ฟเวอร์ (Web Server)

เว็บเซิร์ฟเวอร์ คือ โปรแกรมคอมพิวเตอร์ ซึ่งทำหน้าที่ให้บริการข้อมูลแก่ Client หรือเครื่องคอมพิวเตอร์ที่ขอรับบริการ ในรูปแบบสื่อผสมผ่านระบบเครือข่ายโดยสามารถแสดงผล ผ่านโปรแกรมเว็บเบราว์เซอร์ หรืออาจกล่าวได้ว่าเว็บเซิร์ฟเวอร์คือโปรแกรมที่คอยให้บริการแก่ Client ที่ร้องขอข้อมูลเข้ามาโดยผ่านเว็บเบราว์เซอร์ โปรแกรมที่นิยมนำมาใช้เป็นเครื่องบริการเว็บ ได้แก่ อาปาเช่ (Apache Web Server) และไมโครซอฟท์ไอไอเอส (Microsoft IIS) โดย XAMPP เป็นโปรแกรม Apache Web Server สำหรับจำลองเครื่องคอมพิวเตอร์ส่วนบุคคลให้ทำงานเป็นเว็บเซิร์ฟเวอร์ นั่นคือเครื่องคอมพิวเตอร์จะเป็นทั้งเครื่องแม่และเครื่องลูกในเครื่องเดียวกัน ทำให้ไม่ต้องเชื่อมต่อกับอินเทอร์เน็ต โปรแกรม XAMPP จะมาพร้อมกับภาษา PHP สำหรับพัฒนาเว็บแอปพลิเคชันที่เป็นที่นิยม ฐานข้อมูล MySQL, Apache จะทำหน้าที่เป็นเว็บ เซิร์ฟเวอร์, Perl, OpenSSL และ phpMyadmin โปรแกรม XAMPP จะอยู่ในรูปแบบของไฟล์ Zip, tar, 7z หรือ exe

2.8 โปรแกรม Text Editor

โปรแกรม Text Editor ใช้สำหรับสร้างและแก้ไขข้อความในการสร้างเว็บเพจสามารถแก้ไขโค้ดได้หลายภาษา เช่น HTML, PHP, Java, C/C++, CSS, ASP, Perl, JavaScript และ VBScript เป็นต้น ในการเขียนและแก้ไขตัวอักษรซึ่งเป็นคำสั่งต่าง ๆ ปัจจุบันมีโปรแกรม Text Editor หลายโปรแกรม เช่น Notepad, EditPlus หรือโปรแกรม Dreamweaver ซึ่งมีคุณสมบัติเป็นโปรแกรมสำเร็จรูปในการสร้างเว็บเพจ โดยมี Notepad++ เป็นโปรแกรม Text Editor ที่ใช้กันมานานและมี การอัปเดตคุณสมบัติใหม่ ๆ อยู่ตลอดเวลา มีคุณสมบัติดังต่อไปนี้

- 1) ซอฟต์แวร์ Open Source สามารถดาวน์โหลดติดตั้งใช้งานได้ฟรี
- 2) โปรแกรมมีขนาดเล็ก ติดตั้งง่าย
- 3) มี Syntax Highlighting ซึ่งช่วยให้การเขียนโค้ดโปรแกรมได้สะดวก รองรับ ภาษาคอมพิวเตอร์ได้หลายภาษา เช่น C, C++, Java, C#, XML, HTML, PHP, CSS และอื่นๆ อีกมากมาย
- 4) ระบบเติมคำอัตโนมัติ
- 5) สามารถเปิดเอกสารได้หลายชนิดในเวลาเดียวกัน
- 6) ระบบกรองคำ เพื่อการค้นหาและแทนที่คำ
- 7) รองรับการเข้ารหัสเช่น ASCII และ Unicode (UTF-8) และอื่น ๆ อีกมากมาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.9 ภาษา C

ภาษา C เป็นภาษาคอมพิวเตอร์เพื่อวัตถุประสงค์ทั่วไป เป็นภาษาที่มีความจำเป็นมาก สนับสนุนการเขียนโปรแกรมที่มีโครงสร้าง การกำหนดขอบเขตของตัวแปร และการเรียกใช้ตัวเอง (recursion) และมีความใกล้เคียงกับภาษาในระดับ low level นั่นคือภาษาที่สามารถทำงานได้ดีในระดับของฮาร์ดแวร์ ภาษา C ถูกออกแบบมาให้สามารถที่จะทำงานร่วมกับคำสั่งพื้นฐานของคอมพิวเตอร์ได้อย่างมีประสิทธิภาพ เพราะฉะนั้นจึงถูกพบบ่อยในการใช้สร้างแอปพลิเคชันในสมัยก่อนที่เขียนโดยภาษาแอสเซมบลีรวมถึงระบบปฏิบัติการ เช่นเดียวกับกับซอฟต์แวร์ประยุกต์สำหรับคอมพิวเตอร์ ซุปเปอร์คอมพิวเตอร์และระบบฝังตัว [11]

2.9.1 โครงสร้างของโปรแกรมภาษา C

ในภาษาซีจะเขียนโปรแกรมโดยการเรียกใช้แต่ละชุดของโปรแกรมที่เรียกว่า ฟังก์ชัน (Function) หรือในโปรแกรมภาษาอื่นอาจจะเรียกว่า โปรแกรมย่อยหรือชุดคำสั่งย่อย (Procedure) ฟังก์ชันเหล่านี้จะมีชื่ออะไรก็ได้ ก็ฟังก์ชันก็ได้ แต่อย่างน้อยต้องมี 1 ฟังก์ชันที่ชื่อ main เพื่อให้โปรแกรมเริ่มทำงานที่ฟังก์ชันนี้ แสดงตามตัวอย่างโปรแกรม 2.1 โครงสร้างของโปรแกรมที่เขียนด้วยภาษา C แบ่งย่อยได้เป็น 3 ส่วนดังนี้

- 1) ส่วนเรียกใช้ไฟล์อื่น ๆ เป็นส่วนที่บอกให้คอมไพเลอร์ไปดึงไฟล์อื่นที่กำหนดมาแปลรวมด้วย ไฟล์เหล่านี้อาจจะเป็นไฟล์มาตรฐานที่มีให้แล้วในภาษาซี หรือเป็นไฟล์ที่เขียนขึ้นมาใหม่ก็ได้ โดยส่วนใหญ่มักจะเป็นไฟล์ที่มีนามสกุลเป็น .h
- 2) ส่วนกำหนดชื่อในโปรแกรม เป็นส่วนที่ใช้กำหนดค่าคงที่ ตัวแปร และค่าอื่น ๆ ที่ต้องการ
- 3) ส่วนคำสั่ง จะประกอบด้วยคำสั่งต่าง ๆ หรือฟังก์ชันอื่น ๆ ที่ใช้ในการทำงานของโปรแกรม

โปรแกรม 2.1 แสดงลักษณะของการเขียนโปรแกรมภาษา C

```
#include <stdio.h>
void main(void) {
printf("Hello World");
}
```

2.9.2 ตัวแปรในภาษา C

ตัวแปร หมายถึงชื่อเรียกแทนพื้นที่เก็บข้อมูลในหน่วยความจำ มีชนิดของข้อมูลหรือรูปแบบของตัวแปร แสดงดังตารางที่ 2.3 ซึ่งการกำหนดตัวแปร ทำได้ 2 แบบ คือ

- 1) กำหนดไว้นอกกลุ่มคำสั่งหรือฟังก์ชัน เรียกตัวแปรนี้ว่า Global Variable กำหนดไว้นอกฟังก์ชันใช้งานได้ทั้งโปรแกรม มีค่าเริ่มต้นเป็น 0 (กรณีไม่ได้กำหนดค่าเริ่มต้น)
- 2) กำหนดไว้ในกลุ่มคำสั่งหรือฟังก์ชัน เรียกตัวแปรนี้ว่า Local Variable กำหนดไว้ภายในฟังก์ชัน ใช้งานได้ภายในฟังก์ชันนั้น และไม่ถูกกำหนดค่าเริ่มต้นโดยอัตโนมัติ

ตารางที่ 2.3 แสดงชนิดและขนาดของข้อมูลที่ใช้ในภาษา C (MarcusCode, 2018)

ชนิด	ขนาดความกว้าง	ช่วงของค่า
Char	8 บิต	ASCII character (-128 ถึง 127)
Unsigned char	8 บิต	0-255
Int	16 บิต	-32768 ถึง 32767
long int	32 บิต	-2147483648 ถึง 2147483649
Float	32 บิต	3.4E-38 ถึง 3.4E+38 หรือ ทศนิยม 6 ตำแหน่ง
Double	64 บิต	1.7E-308 ถึง 1.7E+308 หรือ ทศนิยม 12 ตำแหน่ง
Unsigned int	16 บิต	0 ถึง 65535
Unsigned long int	32 บิต	0 ถึง 4294967296

2.9.3 ฟังก์ชันในการรับและแสดงผลข้อมูล

ฟังก์ชันในการรับค่าข้อมูลในภาษา C จะมีฟังก์ชันมาตรฐานในการรับค่าข้อมูล หรือค่าตัวแปรออกมาทางจอภาพ ฟังก์ชันนั้นคือ ฟังก์ชัน scanf ส่วนฟังก์ชันในการแสดงผลข้อมูล จะมีฟังก์ชันมาตรฐานในการแสดงผลข้อมูลหรือค่าตัวแปรออกมาทางจอภาพ ฟังก์ชันนั้นคือ ฟังก์ชัน printf ภายในฟังก์ชันประกอบด้วย 2 ส่วนที่สำคัญคือ

- 1) ส่วน control สามารถแสดงผลเป็นข้อความหรือรหัสรูปแบบโค้ด ดังตารางที่ 2.4
- 2) ส่วน argument list เป็นชุดตัวแปร ค่าคงที่ หรือ นิพจน์ที่ต้องการนำมาแสดงผล ถ้ามีมากกว่า 1 ค่าจะแยกออกจากกัน โดยใช้เครื่องหมายคอมมา (,) นอกจากนี้ ยังมี Escape Sequence คือรหัสพิเศษที่แทรกลงไปในการพิมพ์ที่ตรง เพื่อใช้ควบคุมการแสดงผลของตัวอักษรในลักษณะต่าง ๆ โดยการเขียนจะต้องมีเครื่องหมาย \ (Back-Slash) นำหน้ารหัสควบคุมการแสดงผล ในตารางที่ 2.5

ตารางที่ 2.4 แสดงรหัสรูปแบบที่ใช้ในฟังก์ชัน printf (MarcusCode, 2018)

รหัสรูปแบบ	ชนิดตัวแปร	ลักษณะการแสดงผลออกจอภาพ
%d	int	ใช้แสดงข้อมูลที่เป็นเลขจำนวนเต็มฐานสิบ
%ld	long int	ใช้แสดงข้อมูลที่เป็นเลขจำนวนเต็มฐานสิบแบบ long
%u	unsigned int	ใช้แสดงข้อมูลที่เป็นเลขจำนวนเต็มฐานสิบแบบ unsigned
%c	char	ใช้แสดงข้อมูลที่เป็นตัวอักษร
%s	string	ใช้แสดงข้อมูลที่เป็นตัวแปรสตริงหรือชุดตัวอักษร
%o	int (octal)	ใช้แสดงข้อมูลที่เป็นเลขฐานแปด
%x	int (hexa)	ใช้แสดงข้อมูลที่เป็นเลขฐานสิบหก
%f	float	ใช้แสดงข้อมูลที่เป็นเลขทศนิยม ที่ไม่มีเลขยกกำลัง
%e	float, double	ใช้แสดงข้อมูลที่เป็นเลขทศนิยม ที่มีเลขชี้กำลัง
%lf	double	ใช้แสดงข้อมูลที่เป็นเลขทศนิยมแบบ double

ตารางที่ 2.5 แสดง Escape sequence ต่าง ๆ ที่มีใช้ในภาษา C (MarcusCode, 2018)

Escape Sequence	ค่า	หน้าที่
\a	0x07	เสียงดังออกลำโพงหนึ่งครั้ง
\b	0x08	เลื่อน cursor ไปลบตัวอักษรทางซ้ายมือหนึ่งตัวอักษร
\f	0x0c	ขึ้นหน้าใหม่
\n	0x0a	ขึ้นบรรทัดใหม่
\r	0x0d	เลื่อน cursor ไปทางซ้ายมือสุดของบรรทัด
\t	0x09	ตั้ง tab ในแนวนอน
\v	0x0b	ตั้ง tab ในแนวตั้ง
\\	0x5c	เครื่องหมาย \
\'	0x2c	เครื่องหมาย '
\"	0x22	เครื่องหมาย "
\?	0x3f	เครื่องหมาย ?

2.9.3.1 ฟังก์ชันอื่น ๆ ที่ใช้ในการรับและแสดงข้อมูล

ภาษา C นอกจากจะมีฟังก์ชัน scanf สำหรับรับข้อมูลและฟังก์ชัน printf สำหรับแสดงผลแล้ว ยังมีฟังก์ชันอื่น ๆ อีก ได้แก่

- 1) getchar () เป็นฟังก์ชันที่ใช้รับข้อมูลเข้ามาทางแป้นพิมพ์ทีละ 1 ตัวอักษร โดยต้องกด enter ทุกครั้งเมื่อสิ้นสุดข้อมูล อักษรจะปรากฏบนจอภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับบริการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 2) getch () เป็นฟังก์ชันที่ใช้รับข้อมูลที่เป็นตัวอักษร 1 ตัว เข้ามาทางแป้นพิมพ์ โดยเมื่อป้อน ข้อมูลเสร็จไม่ต้องกดปุ่ม enter และอักขระที่ป้อนเข้ามาจะไม่ปรากฏบนจอภาพ
- 3) gets () เป็นฟังก์ชันที่ใช้รับข้อมูลที่เป็นข้อความ จากแป้นพิมพ์เข้ามาเก็บไว้ในตัวแปร
- 4) putchar () เป็นฟังก์ชันที่ใช้ให้คอมพิวเตอร์แสดงผลบนจอภาพทีละ 1 ตัวอักษร
- 5) puts () เป็นฟังก์ชันที่ใช้แสดงผลข้อมูลที่เป็นข้อความที่เก็บไว้ในตัวแปรชุดออกมาบนจอภาพ

2.9.4 เครื่องหมายดำเนินการ (Operators)

เครื่องหมายดำเนินการ (Operators) หรือ ตัวดำเนินการ นั้นกำหนดการกระทำที่เกิดขึ้นกับตัวแปรและค่าคงที่ โดยที่นิพจน์ประกอบด้วยตัวแปร และค่าคงที่ และใช้ตัวดำเนินการคำนวณเพื่อให้ได้ค่า ตัวดำเนินการในภาษา C มีจำนวนมากเมื่อเปรียบเทียบกับภาษาคอมพิวเตอร์อื่น

2.9.4.1 ตัวดำเนินการทางคณิตศาสตร์

ตัวดำเนินการทางคณิตศาสตร์ เป็นตัวดำเนินการที่คำนวณทางด้านคณิตศาสตร์ให้มีค่าออกมามีชนิดสอดคล้องกับชนิดของตัวถูกกระทำ แสดงดังตารางที่ 2.6

ตารางที่ 2.6 แสดงประเภทของตัวดำเนินการทางคณิตศาสตร์ (MarcusCode, 2018)

เครื่องหมาย	ความหมาย	ตัวอย่าง
+	การบวก	A+B
-	การลบ	A-B
*	การคูณ	A*B
/	การหาร	A/B
%	การหารเอาแต่เศษไว้ (Modulo)	5%3=1 เศษ 2 จะเก็บแต่เศษ 2 เอาไว้
--	การลดค่าลงครั้งละ 1	A -- จะเหมือนกับ A=A-1
++	การเพิ่มค่าขึ้นครั้งละ 1	A++ จะเหมือนกับ A=A+1

2.9.4.2 ตัวดำเนินการสัมพันธ์และตัวดำเนินการตรรกะ

ตัวดำเนินการสัมพันธ์ เป็นเครื่องหมายที่ใช้ในการเปรียบเทียบและตัดสินใจ ซึ่งผลของการเปรียบเทียบจะเป็นได้ 2 กรณีเท่านั้นคือ จริงและเท็จ ภาษาซีถือว่าค่าทางตรรกะจริงและเท็จมีชนิดเป็น int ดังนั้นผลการกระทำทางตรรกะจึงมีค่าเป็นจำนวนเต็ม และมีค่าได้เพียงสองค่าคือ 1 หรือตัวเลขใด ๆ แทนค่าความจริงเป็นจริง และ 0 แทนค่าความจริงเป็นเท็จ ส่วนตัวดำเนินการเอกสารถูกมองไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการคำนวณว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตรรกะ ใช้ในการเปรียบเทียบ และกระทำทางตรรกะกับค่าตัวเลข หรือค่าที่อยู่ในตัวแปร ผลลัพธ์ที่ได้จะมีเพียง 2 ค่าเช่นเดียวกับผลลัพธ์ของตัวดำเนินการสัมพันธ์คือจริงและเท็จ รายละเอียดแสดงดังตารางที่ 2.7

ตารางที่ 2.7 แสดงตัวดำเนินการสัมพันธ์และตัวดำเนินการตรรกะ (MarcusCode, 2018)

Operator	เครื่องหมาย	ความหมาย
ตัวดำเนินการสัมพันธ์ (Relational Operator)	<	น้อยกว่า
	>	มากกว่า
	<=	น้อยกว่า หรือ เท่ากับ
	>=	มากกว่า หรือ เท่ากับ
ตัวดำเนินการเท่ากับ (Equality Operator)	==	เท่ากับ
	!=	ไม่เท่ากับ
ตัวดำเนินการตรรกะ (Logical Operator)	!	นิเสธ
	&&	และ
		หรือ

2.9.4.3 ตัวดำเนินการประกอบ

ตัวดำเนินการประกอบ คือ ตัวดำเนินการที่เป็นรูปแบบย่อของตัวดำเนินการกับตัวแปรที่ถูกดำเนินการ ดังตารางที่ 2.8 โดยตัวดำเนินการประกอบของภาษา C มีทั้งหมด 10 รูปแบบ

ตารางที่ 2.8 แสดงตัวประกอบดำเนินการ (MarcusCode, 2018)

Operator	ตัวอย่าง	เทียบเท่ากับ
+=	a += 2;	a = a + 2
-=	a -= 2;	a = a - 2
*=	a *= 2;	a = a * 2
/=	a /= 2;	a = a / 2
%=	a %= 2;	a = a % 2
>>=	a >>= 2;	a = a >> 2
<<=	a <<= 2	a = a << 2
&=	a &= 2;	a = a & 2
^=	a ^= 2;	a = a ^ 2
=	a = 2;	a = a 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.10 ภาษา PHP

PHP ย่อมาจาก PHP Hypertext Preprocessor แต่เดิมย่อมาจาก Personal Home Page เป็นโปรแกรมภาษาที่ทำงานในลักษณะ ภาษาสคริปต์ฝั่งเซิร์ฟเวอร์ (Server-Side scripting language) คือมีลักษณะของการประมวลผลที่ฝั่งเซิร์ฟเวอร์ หลังจากนั้นก็จะทำการส่งผลลัพธ์ในรูปแบบ HTML กลับไปยังเว็บเบราว์เซอร์ ไฟล์ PHP จะเหมือนกับเอกสาร HTML ทั่วไป เนื่องจากสามารถเขียนแท็กคำสั่งภาษา PHP ลงไปในแท็กของ HTML ได้ [2][12] ดังตัวอย่างโปรแกรม 2.2 ในทางกลับกันก็สามารถเขียนโปรแกรมภาษา HTML แทรกไว้ในภาษา PHP ได้เช่นกัน ซึ่งภาษา PHP มีคุณสมบัติดังต่อไปนี้

- 1) PHP เป็นโปรแกรมภาษาที่เป็นแบบ Open source สามารถดาวน์โหลดได้ฟรี
- 2) มีการแปลภาษาและทำการประมวลผลได้อย่างรวดเร็ว
- 3) PHP สามารถทำงานได้ในระบบปฏิบัติการที่ต่างชนิดกัน เช่น Unix, Windows, Mac OS หรือ Risc OS อย่างมีประสิทธิภาพ เนื่องจาก PHP เป็นสคริปต์ที่ต้องทำงานบนเซิร์ฟเวอร์ ดังนั้นคอมพิวเตอร์สำหรับเรียกใช้คำสั่ง PHP จึงจำเป็นต้องติดตั้งโปรแกรมเว็บเซิร์ฟเวอร์ไว้ด้วย เพื่อให้สามารถประมวลผล PHP ได้
- 4) PHP สามารถทำงานได้ในเว็บเซิร์ฟเวอร์หลายชนิด เช่น Personal Web Server (PWS), Apache, OmniHttpd และ Internet Information Service (IIS)
- 5) PHP สนับสนุนการเขียนโปรแกรมเชิงวัตถุ (Object Oriented Programming)
- 6) PHP มีความสามารถในการทำงานร่วมกับระบบจัดการฐานข้อมูลที่หลากหลาย ซึ่งระบบจัดการฐานข้อมูลที่สนับสนุนการทำงานของ PHP เช่น Oracle, MySQL, FilePro, Solid และ FrontBase เป็นต้น
- 7) PHP อนุญาตให้ผู้ใช้สร้างเว็บไซต์ซึ่งทำงานผ่านโปรโตคอลชนิดต่าง ๆ ได้ เช่น LDAP, IMAP, SNMP, POP3 และ HTTP เป็นต้น
- 8) โค้ด PHP สามารถเขียน และอ่านในรูปแบบของ XML ได้

โปรแกรม 2.2 แสดงลักษณะของการเขียนโปรแกรมภาษา PHP แทรกไว้ใน HTML

```
<HTML>
<BODY>
<?php
    Echo "Hello World";
?>
</BODY>
</HTML>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.11 ภาษา HTML

HTML ย่อมาจาก Hyper Text Markup Language คือภาษาคอมพิวเตอร์ที่ใช้ในการแสดงผลของเอกสารบนเว็บไซต์ ถูกพัฒนาและกำหนดมาตรฐานโดยองค์กร World Wide Web Consortium (W3C) และจากการพัฒนาทางด้านซอฟต์แวร์ของ Microsoft ทำให้ภาษา HTML เป็นอีกภาษาหนึ่งที่ใช้เขียนโปรแกรมได้หรือที่เรียกว่า HTML Application [4]

HTML เป็นภาษาประเภท Markup สำหรับการการสร้างเว็บเพจ สามารถทำได้โดยใช้โปรแกรม Text Editor ต่าง ๆ เช่น Notepad, Editplus หรือจะอาศัยโปรแกรมที่เป็นเครื่องมือช่วยสร้างเว็บเพจ เช่น Microsoft FrontPage, DreamWeaver ซึ่งอำนวยความสะดวกในการสร้างหน้า HTML ส่วนการเรียกใช้งานหรือทดสอบการทำงานของเอกสาร HTML จะใช้โปรแกรมเว็บเบราว์เซอร์ เช่น Microsoft Internet Explorer (IE), Mozilla Firefox, Safari, Opera, และ Netscape Navigator เป็นต้น ในการเขียนโปรแกรมภาษา HTML จะประกอบขึ้นจากแท็กคำสั่งต่าง ๆ แบ่งออกเป็น 4 ส่วน มีโครงสร้างของภาษา HTML ดังนี้ [13]

- 1) <HTML>...</HTML> เป็นคำสั่งแรกที่ต้องมีในภาษา HTML ซึ่งบ่งบอกว่านี่คือภาษา HTML โดยจะอยู่ที่จุดเริ่มต้นของเอกสารและท้ายเอกสารในแต่ละแฟ้ม
- 2) <HEAD>...</HEAD> เป็นส่วนกำหนดรายละเอียดหัวข้อเรื่อง
- 3) <TITLE>...</TITLE> เป็นคำสั่งย่อย กำหนดชื่อหัวข้อเรื่องเว็บเพจที่ต้องการซึ่งจะต้องมีความยาวไม่เกิน 64 ตัวอักษร หัวข้อนี้จะปรากฏที่ Title bar ของเบราว์เซอร์ที่ใช้งานและปรากฏบน Bookmark ของโปรแกรมเบราว์เซอร์ด้วย
- 4) <BODY>...</BODY> ส่วนนี้เป็นส่วนเนื้อหาเว็บเพจที่จะแสดง

2.12 ภาษาจาวาสคริปต์

จาวาสคริปต์ คือภาษาคอมพิวเตอร์สำหรับการเขียนโปรแกรมบนระบบอินเทอร์เน็ตที่กำลังได้รับความนิยมอย่างสูง เป็นภาษาสคริปต์เชิงวัตถุ ซึ่งในการสร้างและพัฒนาเว็บไซต์ ใช้ร่วมกับภาษา HTML เพื่อให้เว็บไซต์มีการเคลื่อนไหว สามารถตอบสนองผู้ใช้งานได้มากขึ้น ซึ่งมีวิธีการทำงานในลักษณะแปลความและดำเนินการไปที่ละคำสั่ง (interpret) หรือเรียกว่าอ็อบเจ็กต์โอเรียนเตด (Object Oriented Programming) ที่มีเป้าหมายในการ ออกแบบและพัฒนาโปรแกรมในระบบอินเทอร์เน็ต สำหรับผู้เขียนด้วยภาษา HTML สามารถทำงานข้ามแพลตฟอร์มได้ โดยทำงานร่วมกับภาษา HTML และภาษา Java ได้ทั้งทางฝั่งไคลเอนต์และทางฝั่งเซิร์ฟเวอร์

จาวาสคริปต์ถูกพัฒนาขึ้นโดย Netscape Communications Corporation โดยใช้ชื่อว่า Live Script ออกมาพร้อมกับ Netscape Navigator 2.0 เพื่อใช้สร้างเว็บเพจโดยติดต่อกับเซิร์ฟเวอร์แบบ Live Wire ต่อมา Netscape จึงได้ร่วมมือกับบริษัทซันไมโครซิสเต็มส์ ปรับปรุงระบบของเบราว์เซอร์เพื่อให้สามารถติดต่อกับภาษาจาวาได้ และได้ปรับปรุง LiveScript ใหม่เมื่อ ปี 2538

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แล้วตั้งชื่อใหม่ว่าจาวาสคริปต์ สามารถทำให้การสร้างเว็บเพจ มีลูกเล่นต่าง ๆ มากมายและยังสามารถโต้ตอบกับผู้ใช้ได้อย่างทันที เช่น การใช้เมาส์คลิกหรือการกรอกข้อความในฟอร์ม เป็นต้น

เนื่องจากจาวาสคริปต์ ช่วยให้ผู้พัฒนา สามารถสร้างเว็บเพจได้ตรงกับความต้องการ และมีความน่าสนใจมากขึ้น ประกอบกับเป็นภาษาเปิด ที่ใครก็สามารถนำไปใช้ได้ ดังนั้นจึงได้รับความนิยมเป็นอย่างสูง มีการใช้งานอย่างกว้างขวาง รวมทั้งได้ถูกกำหนดให้เป็นมาตรฐานโดย ECMA การทำงานของจาวาสคริปต์ จะต้องมีการแปลความคำสั่ง ซึ่งขั้นตอนนี้จะถูกจัดการโดยบราวเซอร์ เรียกว่าเป็น client-side script ดังนั้น จาวาสคริปต์จึงสามารถทำงานได้ เฉพาะบนบราวเซอร์ที่สนับสนุนซึ่งปัจจุบันบราวเซอร์เกือบทั้งหมดสนับสนุนจาวาสคริปต์ อย่างไรก็ตาม สิ่งที่ต้องระวังคือจาวาสคริปต์มีการพัฒนาเป็นเวอร์ชันใหม่ ๆ ออกมาด้วย ดังนั้น ถ้านำโค้ดของเวอร์ชันใหม่ไปรันบนบราวเซอร์รุ่นเก่าที่ยังไม่สนับสนุน ก็อาจจะทำให้เกิดความผิดพลาดได้

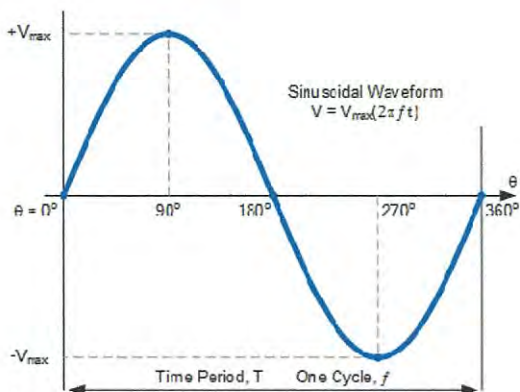
การเขียนคำสั่งของจาวาสคริปต์ ต้องเขียนร่วมกับภาษา HTML โดยแทรกอยู่ภายในระหว่างคำสั่ง <HEAD> กับ </HEAD> หรือจะเขียนหลังจาก <BODY> คำสั่งการเขียนสคริปต์ของ Java Script เริ่มต้นด้วย <script language="JavaScript"> และลงท้ายด้วย </script> ดังตัวอย่างโปรแกรม 2.3

โปรแกรม 2.3 ฟังก์ชัน Hello World เขียนด้วยภาษาจาวาสคริปต์

```
<HTML>
<HEAD>
<script language="JavaScript">
document.write("Hello World!");
</script>
</HEAD>
</HTML>
```

2.13 ไฟฟ้ากระแสสลับ

แรงดันไฟฟ้ากระแสสลับ คือ แรงดันไฟฟ้าไม่คงที่เพราะมีรูปคลื่นสลับไปมาระหว่างบวกและลบ จึงเรียกว่าแรงดันไฟฟ้ากระแสสลับ แสดงดังรูปที่ 2.10 โดยรูปคลื่นของแรงดันไฟฟ้ากระแสสลับนั้นเป็นรูปคลื่นไซน์ ทำให้กระแสไฟฟ้ามักมีการไหลตามการเปลี่ยนแปลงของแรงดันไฟฟ้าไปด้วย [14] ดังรูปที่ 2.11

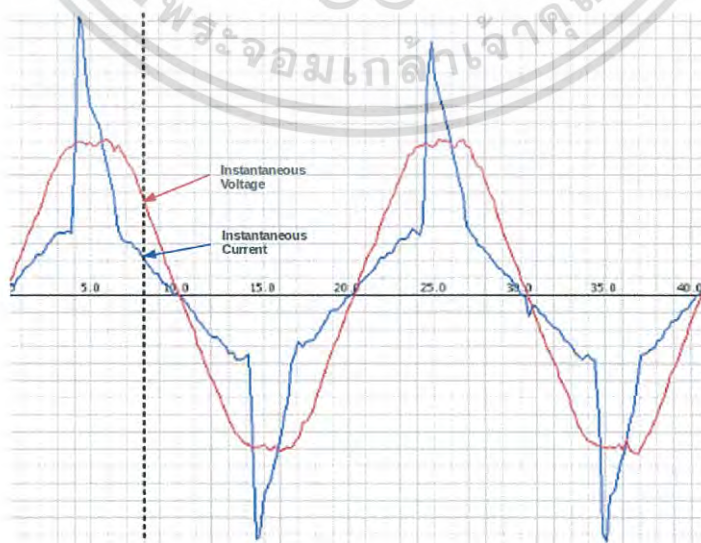


รูปที่ 2.10 รูปคลื่นไซน์ของแรงดันไฟฟ้ากระแสสลับ (Ishan Upadhyay, 2018)

จำนวนครั้งของไซเคิลที่เกิดขึ้นซ้ำ ๆ กัน ใน 1 วินาทีเรียกว่าความถี่ มีหน่วยเป็น เฮิรตซ์ (Hz) ตัวอย่าง เช่น เมื่อกระแสไหลกลับไปกลับมาเป็นจำนวน 50 ครั้งในเวลา 1 วินาทีนั่นคือความถี่ 50 เฮิรตซ์มีความต่อเนื่องและสม่ำเสมอ กระแสและแรงดันสลับมีค่าเปลี่ยนแปลงอยู่เสมอเมื่อเทียบกับเวลา ดังนั้นการบอกค่ากระแสและแรงดันสลับจึงบอกเป็นค่ารากเฉลี่ยกำลังสอง (RMS : Root Mean Square) โดยมีสมการ (2.1) และ (2.2) เมื่อ I_{rms} และ V_{rms} คือค่ากระแสไฟฟ้าสลับและแรงดันไฟฟ้าสลับ ที่ทำให้เกิดปริมาณความร้อนในตัวต้านทาน ที่ต่ออยู่ในวงจร เทียบเท่ากับเมื่อป้อนกระแสและแรงดันไฟตรงค่าเดียวกัน เมื่อ I_m และ V_m คือ ค่ากระแสและแรงดันสูงสุดในหนึ่งไซเคิล

$$I_{rms} = \frac{1}{\sqrt{2}} \times I_m \quad (2.1)$$

$$V_{rms} = \frac{1}{\sqrt{2}} \times V_m \quad (2.2)$$



รูปที่ 2.11 กราฟสุ่มตัวอย่างแรงดันไฟฟ้าและกระแสไฟฟ้าเมื่อทำการวัดอย่างต่อเนื่อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้ (Glyn Hudson, 2016) นี้ ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.13.1 แรงดันไฟฟ้าและกระแสไฟฟ้า RMS

ค่า RMS (Root Mean Square) เป็นการวัดทางสถิติของปริมาณที่มีการเปลี่ยนแปลงอยู่ตลอดเวลา หรือเรียกว่าเป็นรากที่สองของค่าเฉลี่ยของค่านั้น ๆ ยกกำลังสอง ในการวัดแรงดันไฟฟ้าและกระแสไฟฟ้าจะสามารถคำนวณหาค่า RMS เหมือนกัน ดังสมการที่ (2.3)

$$U_{rms} = \sqrt{\frac{\sum_{n=0}^{N-1} u^n(n)}{N}} \quad (2.3)$$

โดย U_{rms} แทน แรงดันไฟฟ้า RMS

$u(n)$ แทน แรงดันไฟฟ้าชั่วขณะที่ทำให้การสุม (Instantaneous voltage)

N แทน จำนวนสุมตัวอย่าง

2.13.2 กำลังไฟฟ้าในระบบไฟฟ้ากระแสสลับ

การเปลี่ยนกระแสไฟฟ้าให้เป็นกำลังไฟฟ้าในวงจรไฟฟ้ากระแสสลับ นั้นจะทำให้เกิดค่ากำลังไฟฟ้าขึ้นมา 3 ค่าด้วยกัน คือ

- 1) กำลังไฟฟ้าจริง (Active Power) กำลังไฟฟ้าจริงคือกำลังไฟฟ้าที่สามารถถูกใช้ในการเปลี่ยนรูปพลังงานและสามารถนำไปใช้ประโยชน์ได้จริง กำลังส่วนนี้มีหน่วยเป็น วัตต์ (Watt) ซึ่งใช้สัญลักษณ์ “P”
- 2) กำลังไฟฟารีแอกทีฟ (Reactive Power) กำลังไฟฟารีแอกทีฟคือกำลังไฟฟ้าที่ไม่ถูกใช้ในการเปลี่ยนรูปพลังงานจึงจัดเป็นกำลังไฟฟ้าที่ไม่พึงประสงค์ กำลังส่วนนี้มีหน่วยเป็น วาร์ (VAR) ซึ่งใช้สัญลักษณ์ “Q”
- 3) กำลังไฟฟ้าปรากฏ (Apparent Power) กำลังปรากฏคือค่ากำลังไฟฟ้าที่เกิดจากผลคูณของกระแสกับแรงดัน มีหน่วยเป็น โวลต์แอมป์ (VA) ซึ่งใช้สัญลักษณ์ “S”

2.13.2.1 การคำนวณกำลังไฟฟ้าจริง

กำลังไฟฟ้าจริง คือ กำลังไฟฟ้าที่จ่ายให้กับโหลดแล้วได้เป็นพลังงานรูปอื่นได้ เช่น ความร้อน แสงสว่าง หรือพลังงานกล ในทางคณิตศาสตร์กำลังไฟฟ้าจริงเกิดจากผลรวมของแรงดันไฟฟ้า กระแสไฟฟ้าและเวลา ดังสมการ (2.4) และ (2.5) ตามลำดับ ทั้งในกรณีช่วงเวลาต่อเนื่องกันและไม่ต่อเนื่องกัน

$$P = \frac{1}{T} \int u(t) \times i(t) dt = U_{rms} \times I_{rms} \times \cos(\varphi) \quad (2.4)$$

โดย U_{rms} แทน แรงดันไฟฟ้า RMS

I_{rms} แทน กระแสไฟฟ้า RMS

$\cos(\varphi)$ แทน ค่าตัวประกอบกำลังไฟฟ้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$P = \frac{1}{N} \sum_{n=0}^{N-1} u(n) \times i(n) \quad (2.5)$$

โดย $u(n)$ แทน แรงดันไฟฟ้าชั่วขณะ
 $i(n)$ แทน กระแสไฟฟ้าชั่วขณะ
 N แทน จำนวนสุ่มตัวอย่าง

2.13.2.2 การคำนวณกำลังไฟฟ้าปรากฏ

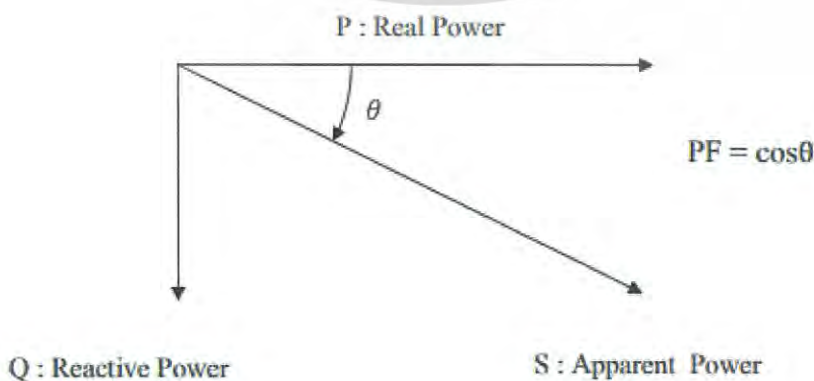
กำลังไฟฟ้าปรากฏ (Apparent Power) คือ ผลคูณของแรงดันไฟฟ้ากับกระแสไฟฟ้า ดังสมการ (2.6) มีหน่วยเป็นโวลต์แอมป์หรือ VA

$$S = U_{rms} \times I_{rms} \quad (2.6)$$

2.13.3 ตัวประกอบกำลังไฟฟ้า

ตัวประกอบกำลังไฟฟ้า (Power Factor) คือ อัตราส่วนระหว่างกำลังไฟฟ้าจริง (watt) กับกำลังไฟฟ้าปรากฏ (VA) ดังสมการ (2.7) ความสัมพันธ์ระหว่างกำลังเฉลี่ยกับกำลังปรากฏจะขึ้นอยู่กับค่า \cos ของมุมต่างเฟสระหว่างกำลังไฟฟ้าจริงกับกำลังไฟฟ้าปรากฏเป็นสำคัญ ดังรูปที่ 2.12 จะเห็นได้ว่ากำลังเฉลี่ยซึ่งสามารถนำไปใช้ประโยชน์ได้จริง จะมีค่าสูงสุดได้ไม่เกินค่ากำลังปรากฏ และจะมีค่าต่ำสุดได้เท่ากับศูนย์ ถ้าหากค่ากำลังเฉลี่ยมีค่าเท่ากับค่ากำลังปรากฏ แสดงว่ากระแสไฟฟ้าที่จ่ายให้โหลดจะถูกใช้ประโยชน์ได้ทั้งหมด แต่ถ้ากำลังเฉลี่ยน้อยกว่าค่ากำลังปรากฏก็แสดงว่ากระแสที่จ่ายให้โหลดถูกใช้ประโยชน์ได้เพียงบางส่วนเท่านั้น โดยส่วนที่เกินจากกำลังงานที่ใช้ทำงานจริง จะเรียกว่ากำลังงานรีแอกทีฟ (Reactive Power) ซึ่งมีหน่วยเป็นวาร์ (VAR) โดยกำลังงานรีแอกทีฟไม่เกิดประโยชน์ต่อระบบเป็นภาระให้กับเครื่องกำเนิดไฟฟ้า หม้อแปลง และสายส่ง

$$\text{Power factor} = \frac{P}{S} = \cos\theta \quad (2.7)$$



รูปที่ 2.12 สามเหลี่ยมกำลังไฟฟ้า (Ekarin Soipetch, 2016)

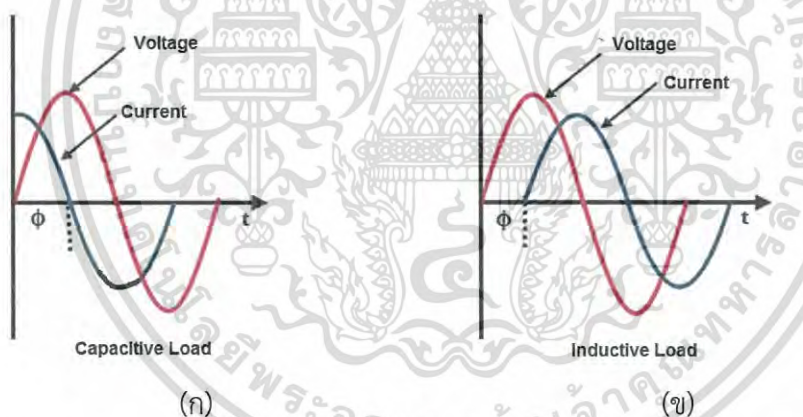
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.13.3.1 ตัวประกอบกำลังไฟฟ้าแบบนำหน้า

โหลดที่มีตัวเก็บประจุ (Capacitor) เป็นองค์ประกอบจะมีใช้น้อยมาก ซึ่ง โหลดประเภทนี้จะทำให้ค่าตัวประกอบกำลังไม่เป็นหนึ่ง เกิดตัวประกอบกำลังไฟฟ้าแบบนำหน้า (Leading) คือ กระแสไฟฟ้าจะนำหน้าแรงดันไฟฟ้า ตามรูปที่ 2.13 (ก) และรูปที่ 2.14 (ก) ส่วนใหญ่ จึงนิยมนำโหลด ประเภทนี้มาปรับปรุงค่าตัวประกอบกำลังไฟฟ้า ของระบบที่มีค่าตัวประกอบ กำลังไฟฟ้าแบบล้าหลัง เพื่อให้ค่าตัวประกอบกำลังไฟฟ้ามีค่าใกล้เคียงหนึ่ง

2.13.3.2 ตัวประกอบกำลังไฟฟ้าแบบล้าหลัง

โหลดประเภทตัวเหนี่ยวนำ (Inductive) จะมีค่าตัวประกอบกำลังไฟฟ้าไม่เป็นหนึ่ง เช่น เครื่องใช้ไฟฟ้าที่ใช้ขดลวด มอเตอร์ บัลลาสต์ของหลอดฟลูออเรสเซนต์ เครื่องปรับอากาศ เป็นต้น จะเห็นได้ว่าหน่วยงานหรือองค์ครส่วนใหญ่ จะหลีกเลี่ยงโหลดประเภทนี้ ไม่ได้และมีเป็นจำนวนมาก ซึ่งจะทำให้ค่าตัวประกอบกำลังไฟฟ้าไม่เป็นหนึ่ง โหลดประเภทนี้จะทำให้ ค่าตัวประกอบกำลังไฟฟ้าล้าหลัง (Lagging) คือ แรงดันไฟฟ้าจะนำหน้ากระแสไฟฟ้า ตามรูปที่ 2.13 (ข) และรูปที่ 3.14 (ข) จำเป็นที่จะต้องปรับปรุงค่าตัวประกอบกำลังไฟฟ้าโดยการนำโหลดประเภทให้ ค่าตัวประกอบกำลังไฟฟ้านำหน้า (Leading) มาต่อเข้าในวงจรไฟฟ้าของระบบ



รูปที่ 2.13 แสดงลักษณะรูปคลื่นระหว่างกระแสไฟฟ้าและแรงดันไฟฟ้า (Simple Projects, 2017)

- (ก) ในกรณีที่โหลดของวงจรไฟฟ้ากระแสสลับเป็นชนิดตัวเก็บประจุ
- (ข) ในกรณีที่โหลดของวงจรไฟฟ้ากระแสสลับเป็นชนิดตัวเหนี่ยวนำ



รูปที่ 2.14 แสดงสามเหลี่ยมกำลังไฟฟ้า (Wikipedia, 2018)

- (ก) ในกรณีที่โหลดชนิดตัวเก็บประจุ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับ (ข) ในกรณีที่โหลดชนิดตัวเหนี่ยวนำ อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

วิธีการดำเนินงาน

ในการดำเนินการโครงการ ได้ทำการกำหนดขั้นตอนการดำเนินการโครงการเพื่อให้บรรลุตามวัตถุประสงค์เพื่อให้สามารถสร้างระบบบันทึกปริมาณกระแสไฟฟ้าผ่านระบบเครือข่ายแบบไร้สายได้อย่างมีประสิทธิภาพ ซึ่งมีขั้นตอนและวิธีการดำเนินงานต่อไปนี้

3.1 การศึกษาเบื้องต้น

ในการสร้างอุปกรณ์เก็บข้อมูลปริมาณการใช้กระแสไฟฟ้าของเครื่องปรับอากาศแล้วนำข้อมูลที่ได้ บันทึกเป็นฐานข้อมูลไว้บนเครื่องแม่ข่าย ขั้นตอนแรกได้ทำการศึกษาค้นคว้าและรวบรวมข้อมูลจากเว็บไซต์ที่เกี่ยวข้องกับโครงการทั้งส่วนที่เป็นซอฟต์แวร์และฮาร์ดแวร์ ลงมือเขียนโปรแกรมจากนั้นประกอบฮาร์ดแวร์ต่าง ๆ ทดสอบโปรแกรมก่อนนำไปใช้งานจริง

3.1.1 การเก็บรวบรวมข้อมูล

ในการเก็บรวบรวมข้อมูลการสร้างอุปกรณ์เก็บข้อมูลปริมาณการใช้กระแสไฟฟ้าของเครื่องปรับอากาศจะเริ่มจากการศึกษาอุปกรณ์ฮาร์ดแวร์และซอฟต์แวร์ที่เกี่ยวข้องกับการสร้างระบบบันทึกปริมาณกระแสไฟฟ้าผ่านระบบเครือข่ายแบบไร้สาย เอกสารและงานวิจัยที่เกี่ยวข้องส่วนใหญ่ได้อ้างอิงมาจากอินเทอร์เน็ต การศึกษาฮาร์ดแวร์จะเก็บข้อมูลเกี่ยวกับคุณสมบัติเฉพาะของอุปกรณ์และการนำไปต่อใช้งาน ส่วนการศึกษาซอฟต์แวร์เพื่อให้สามารถทำการโปรแกรมฮาร์ดแวร์ให้สามารถส่งข้อมูลจากโมดูลเซนเซอร์ไปเก็บไว้ยังเครื่องแม่ข่ายและแสดงผลผ่านเว็บเซิร์ฟเวอร์ได้

3.1.2 ฮาร์ดแวร์ที่ใช้ในการดำเนินงาน

ฮาร์ดแวร์ที่ใช้ในการดำเนินงานประกอบด้วยส่วนหลัก ๆ ทั้งหมด 2 ส่วนคือ ส่วนประมวลผลซึ่งเป็นไมโครคอนโทรลเลอร์และส่วนวัดค่าซึ่งเป็นโมดูลเซนเซอร์ต่าง ๆ ประกอบด้วย

- 1) ไมโครคอนโทรลเลอร์ NodeMCU V3
- 2) โมดูลเซนเซอร์วัดกระแสไฟฟ้า ACS712-20A
- 3) โมดูลเซนเซอร์วัดแรงดันไฟฟ้า ZMPT101B
- 4) โมดูลเซนเซอร์วัดอุณหภูมิและความชื้น AM2302
- 5) โมดูลนาฬิกา DS3231
- 6) ไอซีเบอร์ MCP3008 ใช้ขยายขาสัญญาณอนาล็อก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.3 ซอฟต์แวร์ที่ใช้ในการดำเนินงาน

- 1) โปรแกรม Arduino IDE ใช้ในการเขียนโปรแกรมเพื่อใช้งาน NodeMCU
- 2) โปรแกรม XAMPP ใช้ในการจำลองคอมพิวเตอร์เป็นเว็บเซิร์ฟเวอร์
- 3) โปรแกรม MySQL ใช้เป็นฐานข้อมูลโปรแกรม phpMyAdmin ซึ่งเป็นระบบบริหารฐานข้อมูล MySQL
- 4) โปรแกรม Notepad++ ใช้ในการสร้างเว็บไซต์

3.2 ขั้นตอนการดำเนินงาน

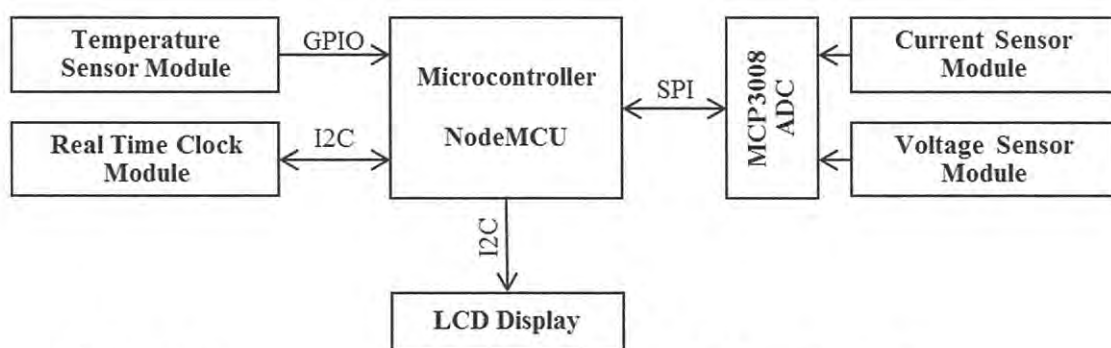
การสร้างระบบบันทึกปริมาณกระแสไฟฟ้าผ่านระบบเครือข่ายแบบไร้สาย เพื่อให้สามารถส่งข้อมูลค่ากระแสไฟฟ้า แรงดันไฟฟ้า อุณหภูมิ ความชื้น และเวลา จากโมดูลต่าง ๆ แล้วนำข้อมูลไปเก็บไว้ที่เครื่องแม่ข่ายผ่านระบบอินเทอร์เน็ต มีขั้นตอนดังนี้

3.2.1 การเก็บรวบรวมข้อมูล

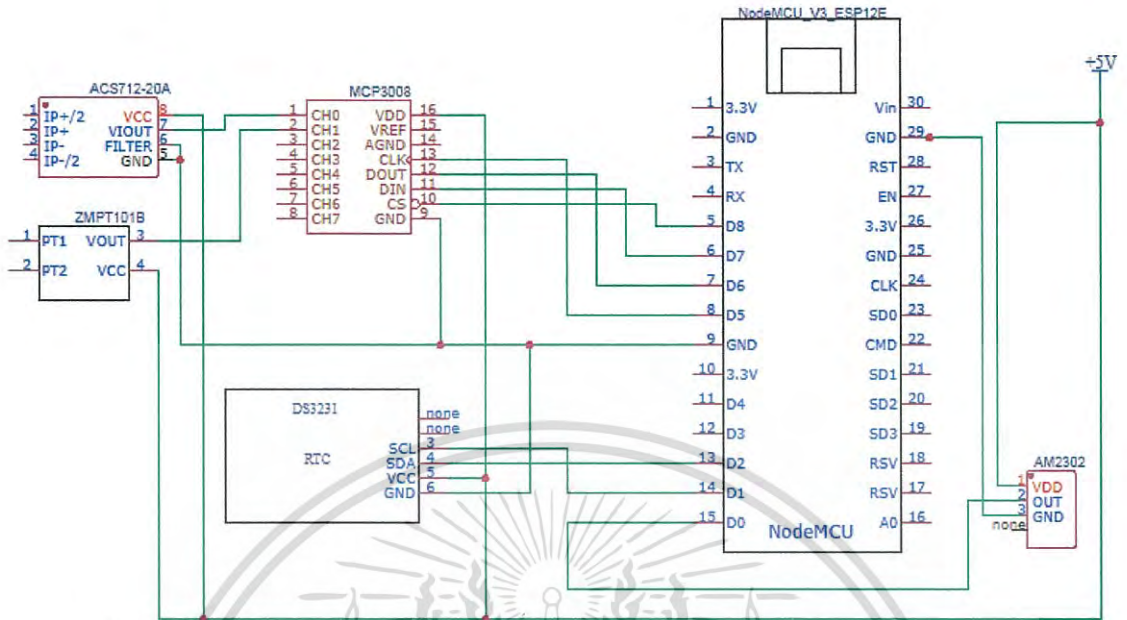
ในการเก็บรวบรวมข้อมูลการสร้างอุปกรณ์เก็บข้อมูลปริมาณการใช้กระแสไฟฟ้าของเครื่องปรับอากาศจะเริ่มจากการศึกษาอุปกรณ์ฮาร์ดแวร์และซอฟต์แวร์ที่เกี่ยวข้องกับการสร้างระบบบันทึกปริมาณกระแสไฟฟ้าผ่านระบบเครือข่ายแบบไร้สาย เอกสารและงานวิจัยที่เกี่ยวข้องส่วนใหญ่ได้อ้างอิงมาจากอินเทอร์เน็ต การศึกษาฮาร์ดแวร์จะเก็บข้อมูลเกี่ยวกับคุณสมบัติเฉพาะของอุปกรณ์และการนำไปต่อใช้งาน ส่วนการศึกษาซอฟต์แวร์เพื่อให้สามารถทำการโปรแกรมฮาร์ดแวร์ให้สามารถส่งข้อมูลจากโมดูลเซนเซอร์ไปเก็บไว้ยังเครื่องแม่ข่ายและแสดงผลผ่านเว็บเซิร์ฟเวอร์ได้

3.2.2 เชื่อมต่ออุปกรณ์ฮาร์ดแวร์

ทำการเชื่อมต่อโมดูลเซนเซอร์วัดกระแสไฟฟ้า ACS712-20A, โมดูลเซนเซอร์วัดแรงดันไฟฟ้า ZMPT101B, โมดูลเซนเซอร์วัดอุณหภูมิและความชื้น AM2302 และโมดูลนาฬิกา DS3231 เข้ากับบอร์ดไมโครคอนโทรลเลอร์ NodeMCU V3 ซึ่งมีรูปแบบการเชื่อมต่อแสดงดังรูปที่ 3.1 และรูปที่ 3.2 แสดงการต่อเซนเซอร์ต่าง ๆ กับขาไมโครคอนโทรลเลอร์



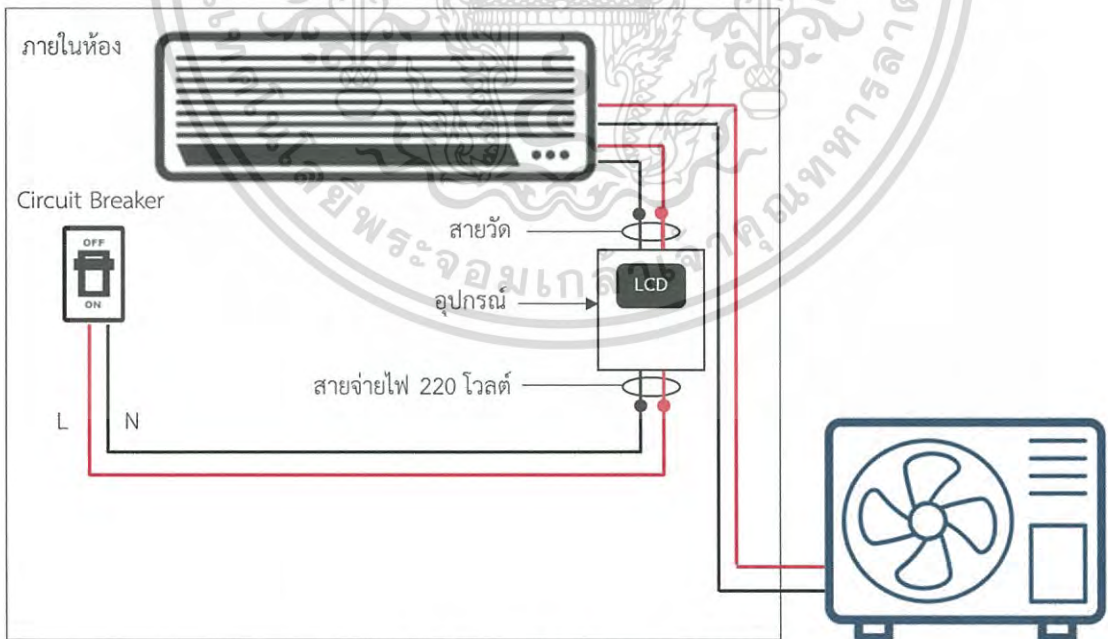
รูปที่ 3.1 บล็อกไดอะแกรมการเชื่อมต่อโมดูลเซนเซอร์เข้ากับไมโครคอนโทรลเลอร์ NodeMCU V3 เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.2 การเชื่อมต่อโมดูลเซนเซอร์ต่าง ๆ เข้ากับไมโครคอนโทรลเลอร์ NodeMCU V3

3.2.3 ติดตั้งอุปกรณ์วัดเข้ากับเครื่องปรับอากาศ

การนำอุปกรณ์ไปใช้งาน ต้องทำการติดตั้งอุปกรณ์วัดเข้ากับสายไฟฟ้าของเครื่องปรับอากาศและคอมเพรสเซอร์ของเครื่องปรับอากาศ ดังรูปที่ 3.3



รูปที่ 3.3 การติดตั้งอุปกรณ์วัดเข้ากับเครื่องปรับอากาศ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

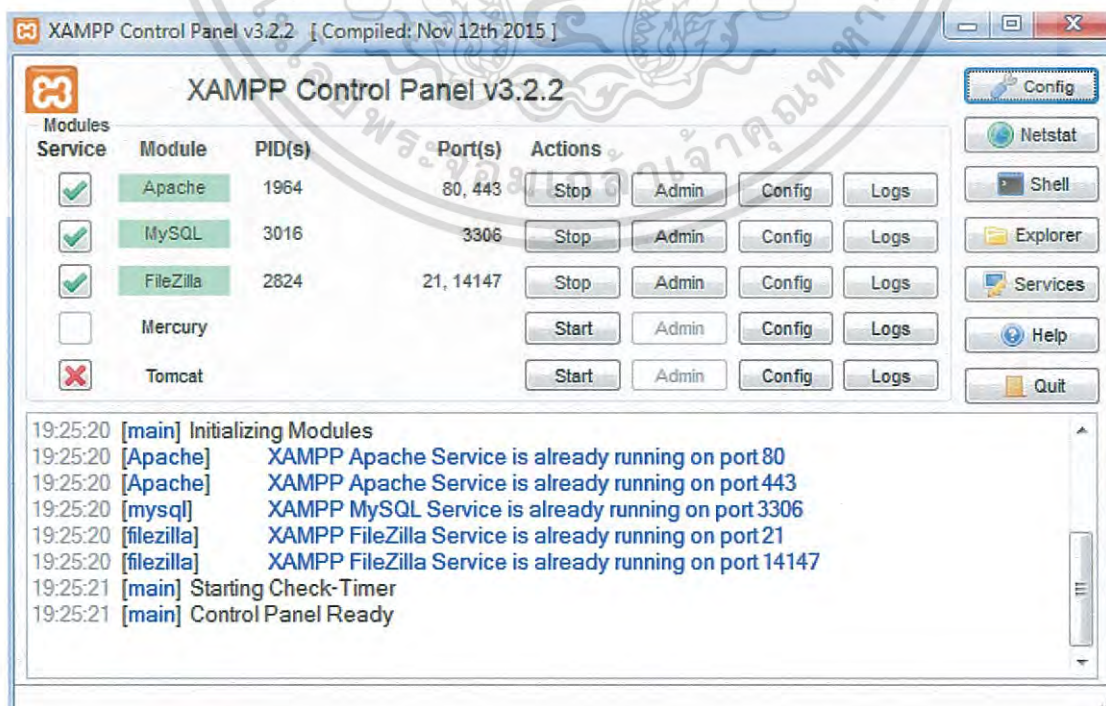
3.2.4 ติดตั้งโปรแกรมซอฟต์แวร์

ติดตั้งโปรแกรม Arduino IDE, MySQL, XAMPP, Notepad++ ลงบนเครื่องคอมพิวเตอร์ เพื่อทำการเขียนโปรแกรม

3.2.5 สร้างฐานข้อมูลบน MySQL

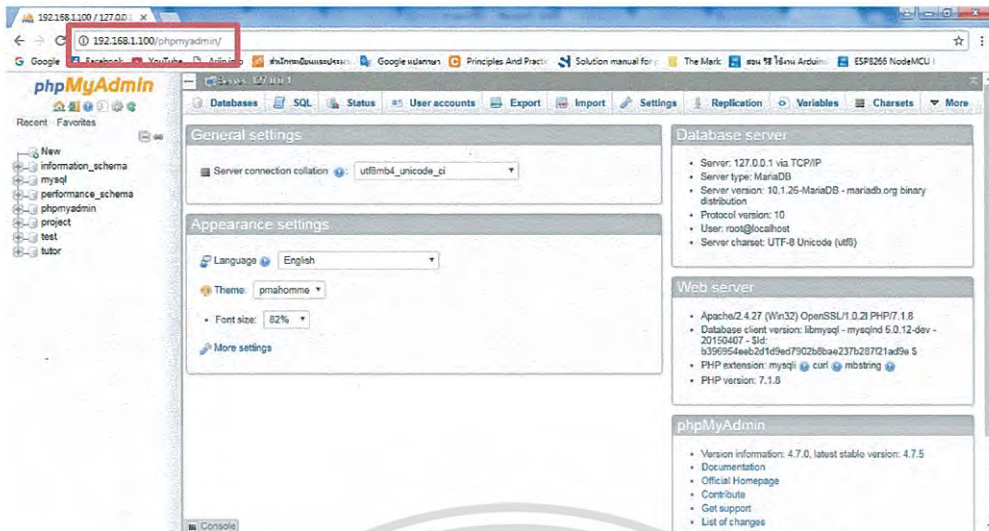
ในการสร้างฐานข้อมูลบน MySQL ต้องใช้โปรแกรม phpMyAdmin ซึ่งเป็นโปรแกรมจัดการฐานข้อมูล MySQL สามารถเรียกใช้โดยการป้อนค่า URL บนเว็บเบราว์เซอร์เป็นค่า IP ของคอมพิวเตอร์ที่จะใช้จำลองเป็นเว็บเซิร์ฟเวอร์จากนั้นจึงสามารถสร้างฐานข้อมูลรูปแบบต่าง ๆ ได้ มีขั้นตอนดังนี้

- 1) เปิดการทำงานของโปรแกรม XAMPP Control Panel ต้องทำการเปิดทำงานโมดูล Apache เพื่อเชื่อมต่อกับเว็บเบราว์เซอร์และเปิดทำงานโมดูล MySQL เพื่อเชื่อมต่อกับเครื่องแม่ข่าย แสดงดังรูปที่ 3.4
- 2) เปิดเว็บเบราว์เซอร์ ใส่ไอพีแอดเดรสบนช่อง URL แสดงดังรูปที่ 3.5 เพื่อเรียกใช้โปรแกรม phpMyAdmin
- 3) สร้างฐานข้อมูลโดยการกดที่ New ตั้งชื่อฐานข้อมูลแล้วกด Create เพื่อสร้างฐานข้อมูล แสดงดังรูปที่ 3.6
- 4) สร้างตารางเพื่อใช้ในการเก็บข้อมูล ตั้งชื่อตารางว่า "data" แสดงดังรูปที่ 3.7
- 5) กำหนดรายละเอียดของตาราง ดังรูปที่ 3.8 ใส่ข้อมูลตามหัวข้อคอลัมน์ที่ต้องการเก็บข้อมูลลงบน MySQL คอลัมน์ที่สร้างขึ้นเพื่อรับข้อมูล ดังรูปที่ 3.9

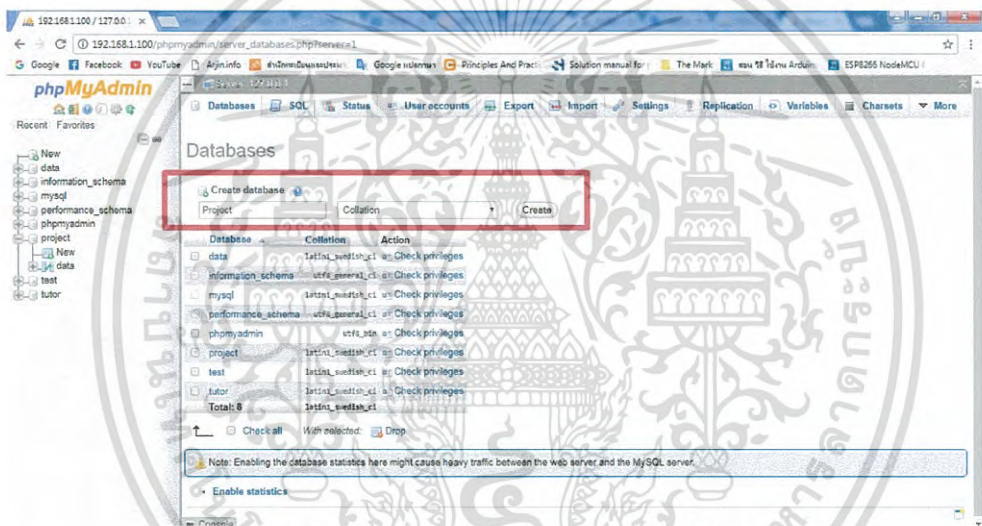


รูปที่ 3.4 แสดงการเปิดใช้งาน Apache และ MySQL บนโปรแกรม XAMPP Control Panel

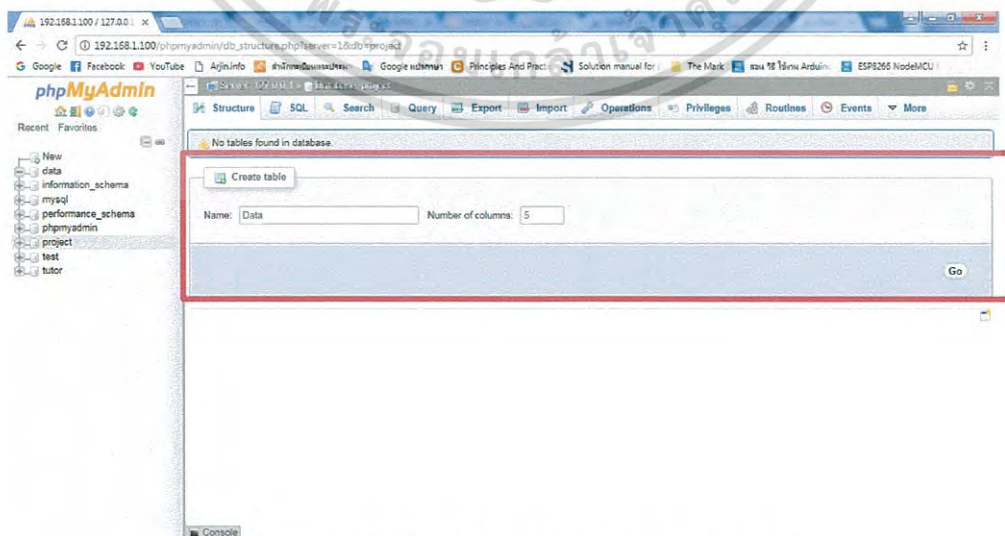
เอกสารนี้เป็นลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี การนำเอกสารนี้ไปใช้โดยไม่ได้รับอนุญาตถือว่าผิดกฎหมาย การทำซ้ำโดยไม่ได้รับอนุญาตถือว่าผิดกฎหมาย การนำเอกสารนี้ไปใช้โดยไม่ได้รับอนุญาตถือว่าผิดกฎหมาย การทำซ้ำโดยไม่ได้รับอนุญาตถือว่าผิดกฎหมาย



รูปที่ 3.5 แสดงการเรียกใช้ phpMyAdmin บนเว็บเบราว์เซอร์

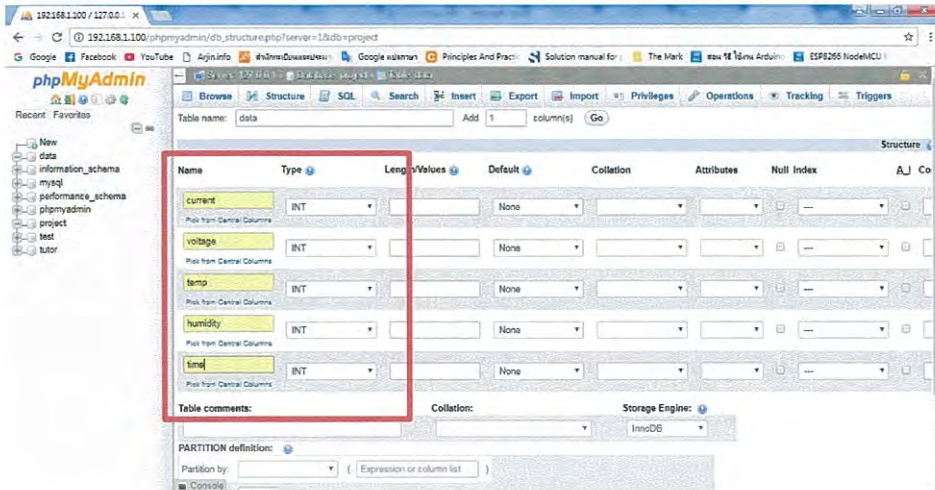


รูปที่ 3.6 แสดงการสร้างฐานข้อมูลบน phpMyAdmin

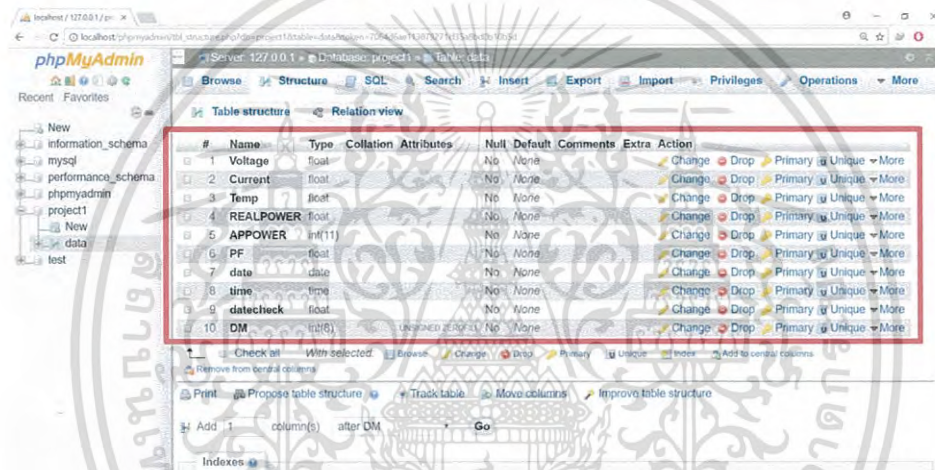


รูปที่ 3.7 แสดงการสร้างตารางบนฐานข้อมูล MySQL

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.8 แสดงการสร้างคอลัมน์ที่ต้องการเก็บค่าบน MySQL



รูปที่ 3.9 แสดงคอลัมน์ที่สร้างขึ้นเพื่อรับข้อมูลจากไมโครคอนโทรลเลอร์เก็บไว้ใน MySQL

3.2.6 สร้างไฟล์ PHP เพื่อเพิ่มข้อมูลแบบ GET ลงใน MySQL

การสร้างไฟล์ .php เป็นเหมือนตัวกลางในการรับค่าจากไมโครคอนโทรลเลอร์ NodeMCU เพื่อส่งข้อมูลไปยังฐานข้อมูล MySQL แสดงดังรูปที่ 3.10 โดยการเขียนโปรแกรมจะใช้ ภาษา PHP เขียนลงในโปรแกรม Notepad

```

add.php - Notepad
File Edit Format View Help
<?php
$temp = $_GET['temp'];

$servername = "localhost";
$username = "root";
$password = "1234";
$dbname = "project";

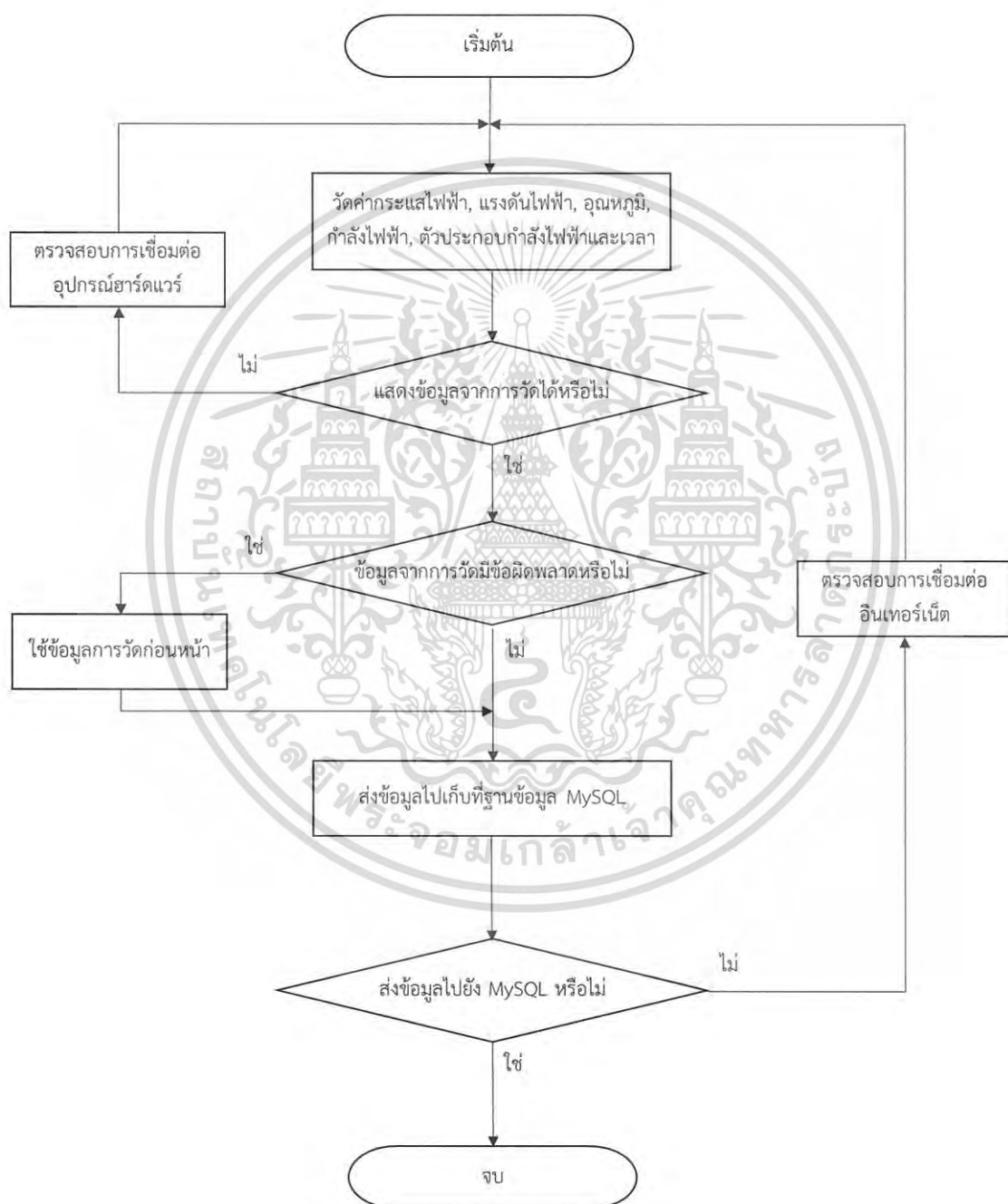
// Create connection
$conn = new mysqli($servername, $username, $password, $dbname);
// Check connection
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}
$val = array($_GET['AmpsRMS'],$_GET['Volt'],$_GET['temp'],$_GET['RealPower'],$_GET['PowerFactor'],$_GET['time'],$_GET['datecheck'],$_GET['monthcheck']);
$sql = "INSERT INTO data(current,voltage,temp,REALPOWER,PF,time,datecheck,MONTH,YEAR) VALUES ($val[0],$val[1],$val[2],$val[3],$val[4],$val[5],$val[6],$val[7],$val[8])";

if ($conn->query($sql) === TRUE) {
    echo "Save OK";
} else
    
```

รูปที่ 3.10 แสดงการเขียนโปรแกรมบน Notepad ด้วย PHP เพื่อติดต่อฐานข้อมูล MySQL

3.2.7 เขียนโปรแกรมให้ NodeMCU ส่งข้อมูลจากโมดูลเซนเซอร์ไปยัง MySQL

NodeMCU สามารถเขียนโปรแกรมผ่านทาง Arduino IDE โดยใช้ภาษา C เพื่อติดต่อกับเซ็นเซอร์ต่าง ๆ การทำงานของโปรแกรมแสดงดังผังงาน (Flowchart) ในรูปที่ 3.11 และส่งข้อมูลผ่านทางระบบไร้สายนำไปแสดงผลผ่านเว็บเบราว์เซอร์โดยการเขียนโปรแกรม ดังรูปที่ 3.12 จะทำให้ส่งข้อมูลจากโมดูลเซนเซอร์ทุกตัวไปเก็บไว้ที่ฐานข้อมูล MySQL ดังรูปที่ 3.13



รูปที่ 3.11 ผังงานแสดงการทำงานของไมโครคอนโทรลเลอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


```

sketch_may23a $
1 #include "EmonLib.h" // Include Emon Library
2 EnergyMonitor emon1; // Create an instance
3 #include <b64.h>
4 #include <HttpClient.h>
5 //-----
6 #include "DHT.h"
7 #define DHTPIN D3
8 #define DHTTYPE DHT22
9 DHT dht(DHTPIN, DHTTYPE);
10 //-----
11 #include <Arduino.h>
12 #include <ESP8266WiFi.h>
13 #include <ESP8266WiFiMulti.h>
14 #include <ESP8266HTTPClient.h>
15 ESP8266WiFiMulti WiFiMulti;
16 //-----
17 #include <Wire.h> // I2C library
18 #include <RtcDS3231.h> // RTC library
19 RtcDS3231<TwoWire> rtcObject(Wire);
20 float VA ;
21 float PF ;
33 NodeMCU 1.0 (ESP-12E Module), 80 MHz, 115200, 4M (3M SPIFFS) on COM7

```

รูปที่ 3.12 แสดงการเขียนโปรแกรมใน Arduino IDE

Voltage	Current	Temp	REALPOWER	APPPOWER	PF	date	time	latecheck	DM
221	0.19	26.3	17	42	0.42	2018-04-22	18:31:34	22	04222018
221	0.28	26.3	37	61	0.6	2018-04-22	18:31:42	22	04222018
221	0.2	26.3	17	45	0.4	2018-04-22	18:31:50	22	04222018
221	0.24	26.3	20	53	0.38	2018-04-22	18:31:54	22	04222018
221	0.21	26.3	18	46	0.4	2018-04-22	18:31:58	22	04222018
221	0.3	26.3	34	65	0.52	2018-04-22	18:32:02	22	04222018
221	0.23	26.3	20	51	0.41	2018-04-22	18:32:06	22	04222018
221	0.2	26.3	17	45	0.39	2018-04-22	18:32:10	22	04222018
221	0.27	26.3	27	58	0.46	2018-04-22	18:32:14	22	04222018
221	0.24	26.3	22	53	0.42	2018-04-22	18:32:18	22	04222018
221	0.28	26.3	35	62	0.57	2018-04-22	18:32:22	22	04222018
221	0.2	26.3	17	45	0.39	2018-04-22	18:32:30	22	04222018
221	0.21	26.3	17	46	0.38	2018-04-22	18:32:34	22	04222018
221	0.24	26.4	23	52	0.44	2018-04-22	18:32:38	22	04222018
221	0.22	26.4	18	47	0.39	2018-04-22	18:32:42	22	04222018
221	0.23	26.4	20	51	0.4	2018-04-22	18:32:46	22	04222018
221	0.26	26.4	25	58	0.43	2018-04-22	18:32:50	22	04222018
221	0.19	26.3	16	42	0.39	2018-04-22	18:32:58	22	04222018
221	0.29	26.4	25	63	0.4	2018-04-22	18:33:02	22	04222018
221	0.21	26.3	18	46	0.4	2018-04-22	18:33:06	22	04222018
221	0.19	26.4	16	42	0.38	2018-04-22	18:33:10	22	04222018
221	0.21	26.4	18	45	0.39	2018-04-22	18:33:14	22	04222018
221	0.22	26.4	18	48	0.38	2018-04-22	18:33:18	22	04222018
221	0.29	26.4	33	64	0.51	2018-04-22	18:33:25	22	04222018
221	0.24	26.4	22	52	0.44	2018-04-22	18:33:29	22	04222018

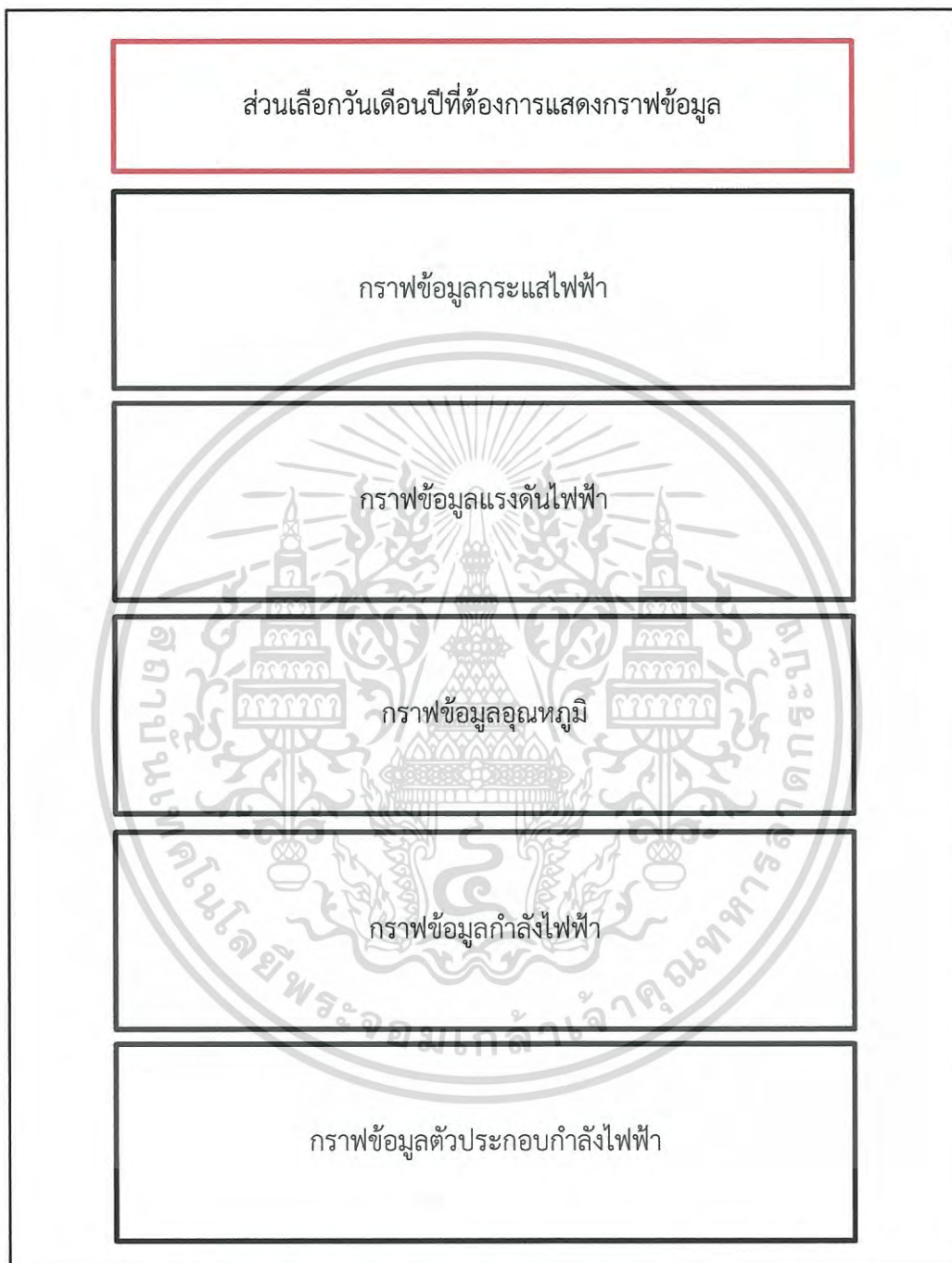
รูปที่ 3.13 แสดงข้อมูลที่วัดได้จากโมดูลเซ็นเซอร์ผ่านหน้าเว็บเซิร์ฟเวอร์ phpMyAdmin

3.2.8 การสร้างเว็บแสดงผล

ในการสร้างเว็บเพจเพื่อแสดงผลข้อมูลที่วัดได้จากโมดูลเซ็นเซอร์ต่าง ๆ ถูกเขียนด้วยซอฟต์แวร์โปรแกรม Notepad++ สะดวก โดยใช้ภาษา C, C++, Java, C#, HTML, PHP, CSS และอื่น ๆ อีกมากมาย แสดงผลออกมาในรูปแบบ HTML โดยส่วนแสดงผลหลักแบ่งออกเป็น 2 ส่วนดังรูปที่ 3.14 ซึ่งมีโครงสร้าง ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 1) ส่วนที่ 1 เป็นส่วนของเมนูให้เลือกช่วงวันเดือนปีที่ต้องการดูกราฟข้อมูล
- 2) ส่วนที่ 2 ส่วนแสดงผลกราฟข้อมูลต่าง ๆ



รูปที่ 3.14 โครงสร้างของเว็บแสดงผล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

ผลการทดลอง

การดำเนินงานของระบบบันทึกปริมาณกระแสไฟฟ้าผ่านระบบเครือข่ายแบบไร้สาย เพื่อศึกษาการ
สร้างอุปกรณ์เก็บข้อมูลปริมาณการใช้กระแสไฟฟ้าของเครื่องปรับอากาศ แล้วนำข้อมูลที่ได้นบันทึกเป็น
ฐานข้อมูลไว้บนเครื่องแม่ข่าย เพื่อนำไปใช้ในการตรวจสอบปริมาณการใช้พลังงานของเครื่องปรับอากาศนั้นได้
ทำทดลองอุปกรณ์กับเครื่องปรับอากาศประเภท Fixed Speed ยี่ห้อ SHARP รุ่น Standard Plasma ขนาด
12,000 บีทียู/ชั่วโมง โดยมีข้อมูลจำเพาะ ดังรูปที่ 4.1



รูปที่ 4.1 ข้อมูลจำเพาะของเครื่องปรับอากาศ SHARP รุ่น Standard Plasma AU-PR13

4.1 การทดสอบส่งข้อมูลจากไมโครคอนโทรลเลอร์ไปเก็บบนฐานข้อมูล MySQL

จากการทดลองอุปกรณ์จะส่งข้อมูลกระแสไฟฟ้า, แรงดันไฟฟ้า, อุณหภูมิ, กำลังไฟฟ้า, ตัวประกอบ
กำลังไฟฟ้า, วันที่และเวลา จากไมโครคอนโทรลเลอร์ NodeMCU V3 ด้วยอัตราสุ่ม (Sampling rate)
ประมาณ 1 วินาที ซึ่งพบว่าข้อมูลที่ปรากฏบนหน้าเว็บเซิร์ฟเวอร์ phpMyAdmin ที่ได้จำลองขึ้น แสดงให้เห็น
ว่าข้อมูลถูกเก็บในฐานข้อมูล MySQL ด้วยระยะห่างทางเวลาประมาณ 1 วินาที ตามรูปที่ 4.2 และรูปที่ 4.3
ขณะเครื่องปรับอากาศทำงานและเครื่องปรับอากาศตัดการทำงาน ตามลำดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Server: 127.0.0.1 » Database: project1 » Table: data

Browse Structure SQL Search Insert Export

+ Options

Voltage	Current	Temp	REALPOWER	APPPOWER	PF	date	time
231	3.64	25.6	826	850	0.97	2018-05-02	05:50:58
231	3.65	25.6	830	852	0.97	2018-05-02	05:50:59
231	3.64	25.6	826	849	0.97	2018-05-02	05:51:01
231	3.65	25.6	830	853	0.97	2018-05-02	05:51:02
231	3.64	25.6	829	851	0.97	2018-05-02	05:51:03
231	3.66	25.6	831	854	0.97	2018-05-02	05:51:04
231	3.66	25.6	830	854	0.97	2018-05-02	05:51:05
231	3.65	25.6	832	854	0.97	2018-05-02	05:51:06
231	3.65	25.6	830	853	0.97	2018-05-02	05:51:08
231	3.67	25.6	834	857	0.97	2018-05-02	05:51:09
231	3.68	25.6	835	859	0.97	2018-05-02	05:51:10
231	3.67	25.6	835	858	0.97	2018-05-02	05:51:11
231	3.68	25.4	836	860	0.97	2018-05-02	05:51:12
231	3.69	25.4	839	861	0.97	2018-05-02	05:51:14
231	3.68	25.6	833	858	0.97	2018-05-02	05:51:15
232	3.7	25.6	841	865	0.97	2018-05-02	05:51:16
232	3.7	25.4	841	866	0.97	2018-05-02	05:51:17
232	3.71	25.4	844	869	0.97	2018-05-02	05:51:18
232	3.71	25.5	844	868	0.97	2018-05-02	05:51:20
232	3.71	25.5	843	868	0.97	2018-05-02	05:51:21
232	3.71	25.6	845	869	0.97	2018-05-02	05:51:22
232	3.72	25.6	846	870	0.97	2018-05-02	05:51:23
232	3.72	25.6	847	870	0.97	2018-05-02	05:51:24
232	3.72	25.6	847	870	0.97	2018-05-02	05:51:25
232	3.73	25.4	850	873	0.97	2018-05-02	05:51:27

รูปที่ 4.2 ตารางข้อมูลที่ถูกเก็บบนฐานข้อมูล MySQL ขณะเครื่องปรับอากาศทำงาน

Server: 127.0.0.1 » Database: project1 » Table: data

Browse Structure SQL Search Insert Export

Voltage	Current	Temp	REALPOWER	APPPOWER	PF	date	time
228	0.11	26.1	8	32	0.25	2018-05-02	01:38:33
228	0.12	26.1	8	33	0.25	2018-05-02	01:38:34
228	0.12	26.1	8	33	0.25	2018-05-02	01:38:35
228	0.12	26.1	8	33	0.25	2018-05-02	01:38:36
228	0.12	26.1	9	34	0.27	2018-05-02	01:38:38
228	0.12	26.1	7	33	0.24	2018-05-02	01:38:39
228	0.12	26.1	8	33	0.24	2018-05-02	01:38:40
228	0.11	26.1	7	32	0.25	2018-05-02	01:38:41
228	0.12	26.1	8	33	0.26	2018-05-02	01:38:42
228	0.12	26.1	8	33	0.26	2018-05-02	01:38:44
228	0.12	26.1	8	33	0.26	2018-05-02	01:38:45
228	0.12	26.1	8	33	0.26	2018-05-02	01:38:46
228	0.11	25.9	8	32	0.26	2018-05-02	01:38:47
228	0.12	25.9	8	33	0.26	2018-05-02	01:38:48
228	0.11	26.1	7	32	0.24	2018-05-02	01:38:50
228	0.11	26.1	8	32	0.27	2018-05-02	01:38:51
228	0.12	26.1	7	33	0.24	2018-05-02	01:38:52
228	0.11	26.1	8	33	0.26	2018-05-02	01:38:53
228	0.12	26.1	7	33	0.23	2018-05-02	01:38:54
228	0.11	26.1	8	32	0.26	2018-05-02	01:38:55
228	0.12	26.2	8	33	0.24	2018-05-02	01:38:57
228	0.11	26.2	8	32	0.25	2018-05-02	01:38:58
228	0.11	26.1	8	32	0.27	2018-05-02	01:38:59
228	0.11	26.1	8	32	0.25	2018-05-02	01:39:00
228	0.11	26.1	8	32	0.26	2018-05-02	01:39:01

รูปที่ 4.3 ตารางข้อมูลที่ถูกเก็บบนฐานข้อมูล MySQL ขณะเครื่องปรับอากาศตัดการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 การทดสอบความถูกต้องของข้อมูล

ในการทดสอบความถูกต้องของอุปกรณ์วัด เพื่อให้อุปกรณ์มีมาตรฐานการวัดที่ตรงกันและสามารถยืนยันผลความถูกต้องของการวัดได้ จะทำการเปรียบเทียบอุปกรณ์วัดกับอุปกรณ์มาตรฐานเพื่อหาความคลาดเคลื่อนของผลการวัดและเปอร์เซ็นต์ความผิดพลาดของอุปกรณ์ ว่าอยู่ในขอบเขตที่เหมาะสมต่อการนำไปใช้งานหรือไม่ ในการทดสอบความถูกต้องของอุปกรณ์จะพิจารณาเฉพาะค่ากระแสไฟฟ้าและแรงดันไฟฟ้าเท่านั้น เพราะโมดูลเซ็นเซอร์วัดอุณหภูมิมีข้อมูลทางเทคนิคที่เชื่อถือได้ ค่ากำลังไฟฟ้าและค่าตัวประกอบกำลังไฟฟ้าเป็นค่าที่ได้จากสมการคำนวณที่มีตัวแปรค่ากระแสไฟฟ้าและแรงดันไฟฟ้า

4.2.1 การทดสอบความถูกต้องของการวัดกระแสไฟฟ้า

การหาความคลาดเคลื่อนและเปอร์เซ็นต์ความผิดพลาดของการวัดกระแสไฟฟ้า จะทำการสุ่มตัวอย่างมาทำการทดสอบ โดยใช้ข้อมูลการวัดกระแสไฟฟ้าที่ถูกเก็บในฐานข้อมูล MySQL เปรียบเทียบกับอุปกรณ์มาตรฐานดิจิทัลแคลมป์มิเตอร์ ยี่ห้อ KYORITSU โดยเก็บข้อมูลทุก ๆ 1 นาที เป็นระยะเวลาทั้งหมด 10 นาที จากการทดลองวัดค่ากระแสไฟฟ้าของอุปกรณ์พบว่ามีความคลาดเคลื่อนและเปอร์เซ็นต์ความผิดพลาดน้อยเมื่อเทียบกับอุปกรณ์มาตรฐานดิจิทัลแคลมป์มิเตอร์ ยี่ห้อ KYORITSU ถือว่าข้อมูลกระแสไฟฟ้าจากอุปกรณ์วัดมีความน่าเชื่อถือเนื่องจากมีเปอร์เซ็นต์ความผิดพลาดประมาณ 1.7768% ดังตารางที่ 4.1

ตารางที่ 4.1 ผลการทดลองเปรียบเทียบความถูกต้องของการวัดกระแสไฟฟ้ากับอุปกรณ์มาตรฐาน

กระแสไฟฟ้าจากฐานข้อมูล MySQL เมื่อใช้อุปกรณ์วัดที่สร้างขึ้น (A)	กระแสไฟฟ้าจากดิจิทัลแคลมป์มิเตอร์ ยี่ห้อ KYORITSU (A)	ค่าความคลาดเคลื่อน (A)	เปอร์เซ็นต์ความผิดพลาดของอุปกรณ์ (%)
3.02	3.04	0.02	0.6579
3.5	3.53	0.03	0.8499
3.8	3.8	0	0
3.8	4.06	0.26	6.4039
4	4.1	0.1	2.4390
4.15	4.23	0.08	1.8913
4.22	4.32	0.1	2.3148
4.31	4.35	0.04	0.9195
4.32	4.35	0.03	0.6897
4.3	4.37	0.07	1.6018
	ค่าเฉลี่ย	0.073	1.7768

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.2 การทดสอบความถูกต้องของการวัดแรงดันไฟฟ้า

การหาความคลาดเคลื่อนและเปอร์เซ็นต์ความผิดพลาดของการวัดแรงดันไฟฟ้า จะทำการสุ่มตัวอย่างมาทำการทดสอบ โดยใช้ข้อมูลการวัดแรงดันไฟฟ้าที่ถูกเก็บในฐานข้อมูล MySQL เปรียบเทียบกับอุปกรณ์มาตรฐานดิจิทัลแคลมป์มิเตอร์ ยี่ห้อ KYORITSU โดยเก็บข้อมูลทุก ๆ 1 นาที เป็นระยะเวลาทั้งหมด 10 นาที จากการทดลองวัดค่าแรงดันไฟฟ้าของอุปกรณ์พบว่ามีความคลาดเคลื่อนและเปอร์เซ็นต์ความผิดพลาดน้อยมากเมื่อเทียบกับอุปกรณ์มาตรฐานดิจิทัลแคลมป์มิเตอร์ ยี่ห้อ KYORITSU ถือว่าข้อมูลแรงดันไฟฟ้าจากอุปกรณ์วัดมีความน่าเชื่อถือเนื่องจากมีเปอร์เซ็นต์ความผิดพลาดประมาณ 0.1732% ดังตารางที่ 4.2

ตารางที่ 4.2 ผลการทดลองเปรียบเทียบความถูกต้องของการวัดแรงดันไฟฟ้ากับอุปกรณ์มาตรฐาน

แรงดันไฟฟ้าจากฐานข้อมูล MySQL เมื่อใช้อุปกรณ์วัดที่สร้างขึ้น (V)	แรงดันไฟฟ้าจากดิจิทัลแคลมป์มิเตอร์ ยี่ห้อ KYORITSU (V)	ค่าความคลาดเคลื่อน (V)	เปอร์เซ็นต์ความผิดพลาดของอุปกรณ์ (%)
231	230	1	0.4348
231	231	0	0
232	231	1	0.4329
232	231	1	0.4329
232	232	0	0
232	232	0	0
233	232	1	0.4310
233	233	0	0
234	234	0	0
234	234	0	0
	ค่าเฉลี่ย	0.4	0.1732

4.3 การแสดงผลกราฟข้อมูลบนเว็บเบราว์เซอร์

การแสดงผลกราฟข้อมูลบนเว็บเบราว์เซอร์จะปรากฏโครงสร้างหลัก 2 ส่วน คือ ส่วนของเมนูให้เลือกช่วงวันเดือนปีที่ต้องการดูกราฟข้อมูล ตามรูปที่ 4.4 ต้องทำการเลือกช่วงวันที่ที่ต้องการก่อน จากนั้นกดปุ่ม Submit เพื่อแสดงผลกราฟข้อมูล พบว่าเมื่อกดปุ่ม Submit เพื่อแสดงกราฟข้อมูล กราฟจะปรากฏบนหน้าเว็บเพจเมื่อเวลาผ่านไป 30 วินาที ถือว่ามีความล่าช้าผิดปกติ อาจเกิดจากการที่ระบบต้องสร้างภาพกราฟิก (Rendering) จำนวนมาก ซึ่งเกี่ยวข้องกับประสิทธิภาพของเครื่องคอมพิวเตอร์ที่ใช้ และส่วนที่สองคือส่วนแสดงผลกราฟข้อมูลต่าง ๆ เมื่อทำการเลือกช่วงวันเดือนปีที่ต้องการ จะปรากฏกราฟข้อมูล 5 กราฟ ได้แก่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 1) กราฟกระแสไฟฟ้า โดยกระแสไฟฟ้าของเครื่องปรับอากาศเมื่อเปิดเครื่องจะมีกระแสสตาร์ท 2-3 เท่าของกระแสปกติซึ่งเกิดขึ้นภายในระยะเวลาสั้น ๆ เพราะคอมเพรสเซอร์เป็นโหลดประเภทมอเตอร์ เนื่องจากเครื่องปรับอากาศที่ใช้เป็นประเภท Fixed Speed คอมเพรสเซอร์จะทำงานไประยะเวลาหนึ่งเพื่อให้อุณหภูมิห้องตรงตามค่าที่ตั้งไว้ แล้วทำการตัดการทำงานและเริ่มทำงานใหม่ทุก ๆ ระยะเวลาประมาณ 6 นาที จากกราฟกระแสไฟฟ้า ตามรูปที่ 4.5 จะพบว่ากระแสสตาร์ทของเครื่องปรับอากาศจะขาดหายไปบางช่วง เป็นผลมาจากอุปกรณ์วัดสามารถส่งข้อมูลด้วยอัตราสุ่ม (Sampling rate) ทุก ๆ เวลาประมาณ 1 วินาที ทำให้ข้อมูลการเกิดกระแสสตาร์ทที่เกิดขึ้นในเวลาสั้น ๆ นั้นมีบางช่วงไม่สามารถส่งข้อมูลแล้วบันทึกลงในฐานข้อมูลได้
- 2) กราฟแรงดันไฟฟ้า ตามรูปที่ 4.6 แรงดันไฟฟ้าของเครื่องปรับอากาศจะมีค่าค่อนข้างคงที่ตลอดระยะเวลาการทำงาน ไม่ว่าจะในช่วงทำงานหรือตัดการทำงานของคอมเพรสเซอร์
- 3) กราฟอุณหภูมิ ตามรูปที่ 4.7 เป็นอุณหภูมิห้องขณะเครื่องปรับอากาศทำงาน พบว่าเมื่อเครื่องปรับอากาศเริ่มทำงาน อุณหภูมิห้องจะลดลงเรื่อย ๆ จนถึงค่าอุณหภูมิที่ทำการปรับตั้งไว้ที่ประมาณ 25 °C
- 4) กราฟกำลังไฟฟ้า จะแสดงทั้งกำลังไฟฟ้าจริงและกำลังไฟฟ้าปรากฏที่ได้จากการคำนวณค่ากระแสไฟฟ้าและแรงดันไฟฟ้า โดยกำลังไฟฟ้าปรากฏจะมีค่ามากกว่ากำลังไฟฟ้าจริง จากกราฟดังรูปที่ 4.8 จะเห็นว่าเครื่องปรับอากาศนั้นมีการทำงานทั้งหมด 3 ช่วง ได้แก่ ช่วง A เป็นช่วงที่เครื่องปรับอากาศทำงาน จากข้อมูลในรูปที่ 4.9 จะเห็นได้ว่าค่ากำลังไฟฟ้าจริงจะใกล้เคียงกับกำลังไฟฟ้าปรากฏมาก ช่วง B เป็นช่วงที่เครื่องปรับอากาศตัดการทำงาน จากข้อมูลในรูปที่ 4.10 จะเห็นได้ว่าค่ากำลังไฟฟ้าจริงมีค่าต่ำมากเมื่อเทียบกับค่ากำลังไฟฟ้าปรากฏ และช่วง C เป็นช่วงที่เกิดกระแสสตาร์ท จากข้อมูลในรูปที่ 4.11 จะมีกำลังไฟฟ้าปรากฏสูงกว่าช่วงที่เครื่องปรับอากาศทำงานปกติ
- 5) กราฟตัวประกอบกำลังไฟฟ้า เป็นกราฟที่ได้จากอัตราส่วนระหว่างกำลังไฟฟ้าจริงกับกำลังไฟฟ้าปรากฏ ตามรูปที่ 4.12 ในช่วงที่เครื่องปรับอากาศทำงานจะมีค่าตัวประกอบกำลังไฟฟ้าใกล้เคียง 1 ซึ่งอาจเป็นผลมาจากการปรับปรุงค่าตัวประกอบกำลังไฟฟ้า ดังตัวอย่างข้อมูลในรูปที่ 4.9 ที่เวลา 01:53:29 จะมีกำลังไฟฟ้าจริงเท่ากับ 878 W และกำลังไฟฟ้าปรากฏเท่ากับ 891 VA ทำให้มีค่าตัวประกอบกำลังไฟฟ้าเท่ากับ 0.99 เพราะกำลังไฟฟ้าจริงกับกำลังไฟฟ้าปรากฏมีค่าใกล้เคียงกัน ในช่วงที่เครื่องปรับอากาศตัดการทำงานจะมีมอเตอร์พัดลมภายในทำงานอยู่ ซึ่งเป็นโหลดประเภทตัวเหนี่ยวนำและมีค่ากระแสไฟฟ้าต่ำ ทำให้ค่าตัวประกอบกำลังมีค่าต่ำ อันเนื่องมาจากไม่มีการปรับปรุงค่าตัวประกอบกำลังในอุปกรณ์ที่กินกระแสไฟฟ้าต่ำ ดังตัวอย่างข้อมูลในรูปที่ 4.10 ที่เวลา 01:54:28 จะมีกำลังไฟฟ้าจริงเท่ากับ 8 W และกำลังไฟฟ้าปรากฏเท่ากับ 33 VA ทำให้มีค่าตัวประกอบกำลังไฟฟ้าเท่ากับ 0.24 เพราะกำลังไฟฟ้าจริงแตกต่างกับกำลังไฟฟ้าปรากฏมาก และในช่วงที่เกิดกระแสสตาร์ท ตามข้อมูลในรูปที่ 4.11 ที่เวลา 01:57:04 จะมี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กำลังไฟฟ้าจริงเท่ากับ 764 W และกำลังไฟฟ้าปรากฏเท่ากับ 1887 VA ทำให้มีค่าตัวประกอบกำลังไฟฟ้าเท่ากับ 0.4

THE CURRENT LOGGING SYSTEM WITH WIRELESS LAN

EXAMPLE : Air Conditionner Sharp 12,000 BTU/Hr with 21.5 sq.m room size

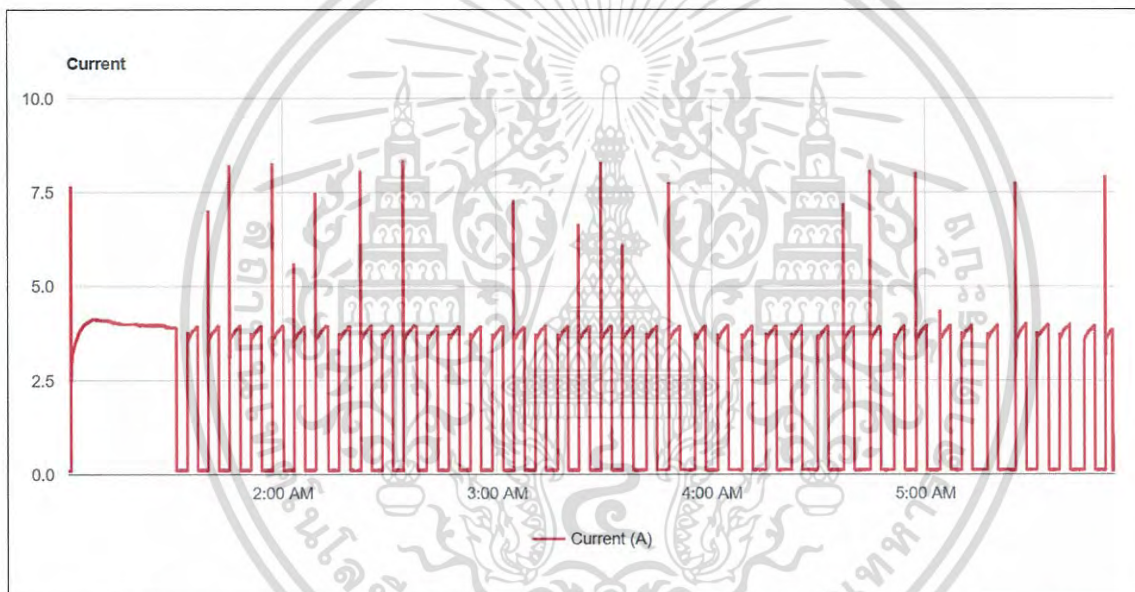
Mon May 21 2018 13:58:55 GMT+0700 (SE Asia Standard Time)

Start Date To

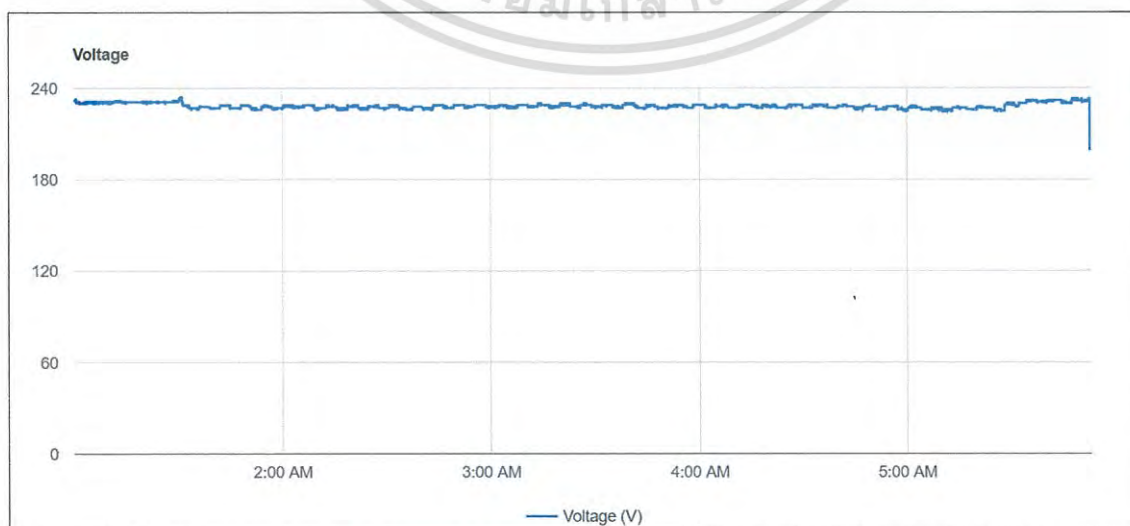
EXAMPLE FOR FILLING DATE : APRIL 22 2018 FILL AS 04222018

[Current](#) : [Voltage](#) : [Temperature](#) : [Power](#) : [Power Factor](#)

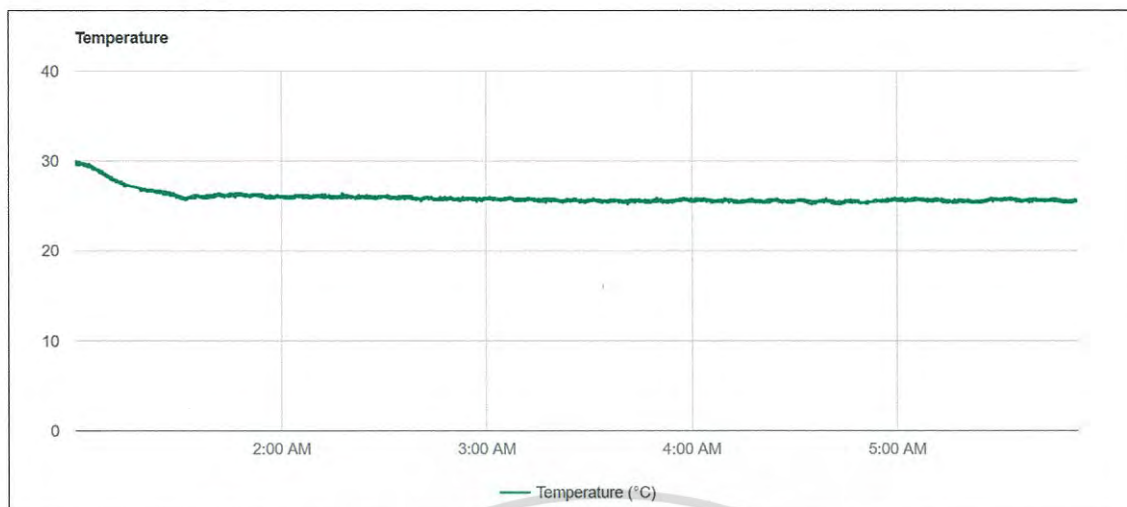
รูปที่ 4.4 แสดงหน้าเว็บเพจเพื่อเลือกช่วงวันที่ที่ต้องการ



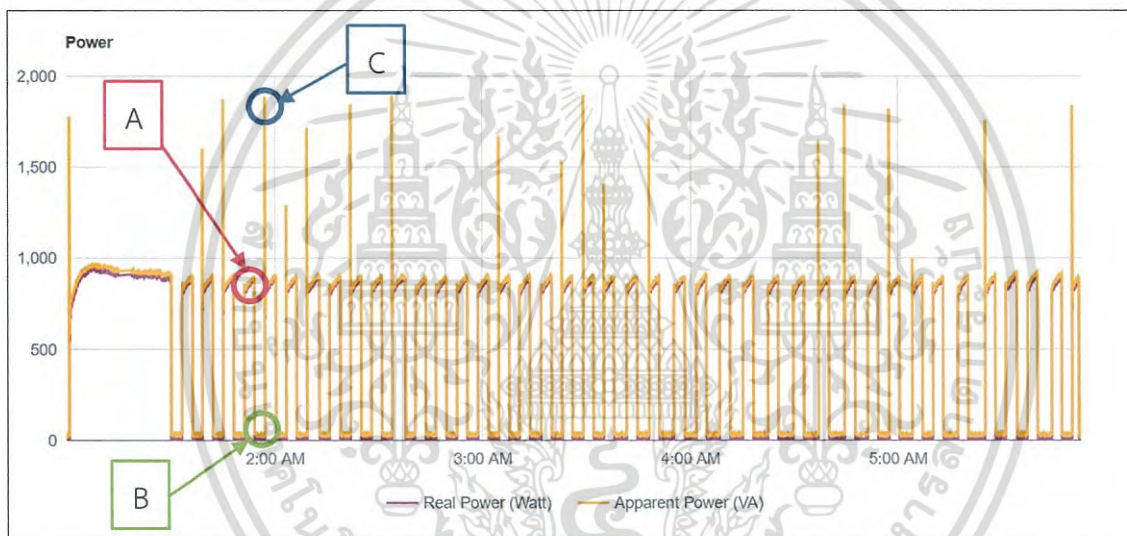
รูปที่ 4.5 แสดงกราฟการวัดกระแสไฟฟ้าของเครื่องปรับอากาศในช่วงวันที่ 2 พฤษภาคม 2561



รูปที่ 4.6 แสดงกราฟการวัดแรงดันไฟฟ้าของเครื่องปรับอากาศในช่วงวันที่ 2 พฤษภาคม 2561 ด้านการคำนวณการสูญเสียพลังงานในระบบปรับอากาศและต้นทุนการดำเนินงานในการดำเนินงานด้านพลังงาน
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.7 แสดงกราฟการวัดอุณหภูมิห้องในช่วงวันที่ 2 พฤษภาคม 2561



รูปที่ 4.8 แสดงกราฟค่ากำลังไฟฟ้าในช่วงวันที่ 2 พฤษภาคม 2561

phpMyAdmin

Server: 127.0.0.1 » Database: project1 » Table: data

Browse Structure SQL Search Insert Export

+ Options

Options	Voltage	Current	Temp	REALPOWER	APPOWER	PF	date	time
	226	3.9	26.2	878	891	0.99	2018-05-02	01:53:29
	226	3.9	26.2	876	891	0.98	2018-05-02	01:53:31
	226	3.89	26.2	876	889	0.98	2018-05-02	01:53:32
	226	3.9	26.2	874	889	0.98	2018-05-02	01:53:33
	226	3.9	26.2	878	891	0.98	2018-05-02	01:53:34
	226	3.9	26.2	877	891	0.99	2018-05-02	01:53:35
	226	3.91	26.2	879	893	0.98	2018-05-02	01:53:36
	226	3.91	26.2	879	893	0.98	2018-05-02	01:53:38
	226	3.91	26.2	878	892	0.98	2018-05-02	01:53:39
	226	3.91	26.2	879	893	0.98	2018-05-02	01:53:40
	226	3.91	26.2	881	894	0.99	2018-05-02	01:53:41
	226	3.91	26.1	880	894	0.98	2018-05-02	01:53:42
	226	3.91	26.1	880	894	0.98	2018-05-02	01:53:43
	226	3.91	26.2	880	894	0.98	2018-05-02	01:53:45
	226	3.92	26.2	882	895	0.99	2018-05-02	01:53:46

รูปที่ 4.9 ตารางข้อมูลที่ถูกเก็บบนฐานข้อมูล MySQL ขณะเครื่องปรับอากาศทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่โดยไม่ได้รับอนุญาต
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Server: 127.0.0.1 » Database: project1 » Table: data

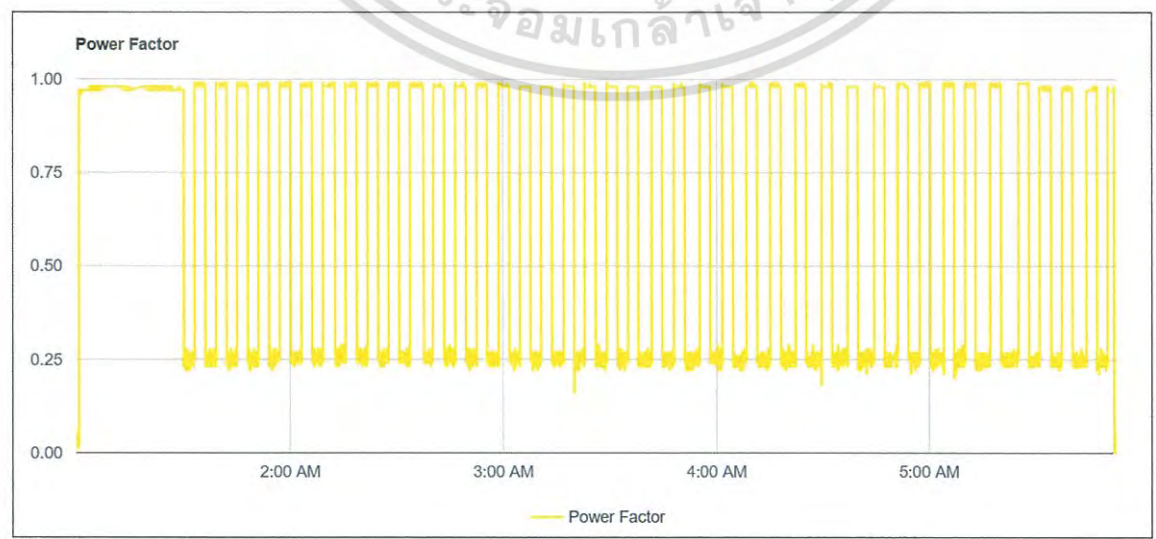
Options	Voltage	Current	Temp	REALPOWER	APPOWER	PF	date	time
	228	0.12	26.2	8	33	0.24	2018-05-02	01:54:28
	228	0.11	26.1	8	32	0.26	2018-05-02	01:54:30
	228	0.11	26.1	7	32	0.23	2018-05-02	01:54:31
	228	0.12	26.1	8	33	0.27	2018-05-02	01:54:32
	228	0.11	26.1	8	33	0.25	2018-05-02	01:54:33
	228	0.11	26.1	8	32	0.26	2018-05-02	01:54:34
	228	0.12	26.1	8	33	0.26	2018-05-02	01:54:35
	228	0.11	26.2	8	32	0.25	2018-05-02	01:54:37
	228	0.11	26.2	7	33	0.24	2018-05-02	01:54:38
	228	0.11	26.1	8	33	0.25	2018-05-02	01:54:39
	228	0.11	26.1	8	32	0.25	2018-05-02	01:54:40
	228	0.11	26.1	8	32	0.25	2018-05-02	01:54:41
	228	0.11	26.1	8	32	0.25	2018-05-02	01:54:42
	228	0.11	26.1	7	32	0.24	2018-05-02	01:54:44
	228	0.11	26.1	8	32	0.26	2018-05-02	01:54:45

รูปที่ 4.10 ตารางข้อมูลที่ถูกเก็บบนฐานข้อมูล MySQL ขณะเครื่องปรับอากาศตัดการทำงาน

Server: 127.0.0.1 » Database: project1 » Table: data

Options	Voltage	Current	Temp	REALPOWER	APPOWER	PF	date	time
	228	0.12	26	8	33	0.25	2018-05-02	01:56:56
	228	0.12	26	8	33	0.25	2018-05-02	01:56:57
	228	0.11	26	8	32	0.26	2018-05-02	01:56:58
	228	0.11	26	8	33	0.25	2018-05-02	01:56:59
	228	0.11	26	8	33	0.25	2018-05-02	01:57:01
	228	0.12	26	8	33	0.25	2018-05-02	01:57:02
	228	0.12	26	8	33	0.26	2018-05-02	01:57:03
	226	8.29	26	764	1887	0.4	2018-05-02	01:57:04
	227	3.01	26.1	677	691	0.98	2018-05-02	01:57:05
	227	3.35	26.1	757	768	0.99	2018-05-02	01:57:07
	227	3.43	26	777	787	0.99	2018-05-02	01:57:08
	227	3.55	26	803	814	0.99	2018-05-02	01:57:09
	227	3.64	26	824	834	0.99	2018-05-02	01:57:10
	227	3.73	26	844	855	0.99	2018-05-02	01:57:11
	227	3.76	26.1	851	862	0.99	2018-05-02	01:57:13

รูปที่ 4.11 ตารางข้อมูลที่ถูกเก็บบนฐานข้อมูล MySQL ขณะเกิดกระแสตาร์ท



รูปที่ 4.12 แสดงกราฟค่าตัวประกอบกำลังไฟฟ้าในช่วงวันที่ 2 พฤษภาคม 2561

เอกสารนี้เป็นลิขสิทธิ์สงวนไว้เพื่อการใช้งานภายในเท่านั้น ไม่อนุญาตให้เผยแพร่โดยไม่ขออนุญาต
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปผลการทดลองและข้อเสนอแนะ

5.1 สรุปผลการทดลอง

โครงการนี้จัดทำขึ้นเพื่อสร้างอุปกรณ์การวัดที่สามารถส่งข้อมูลปริมาณกระแสไฟฟ้า แรงดันไฟฟ้า กำลังไฟฟ้า ค่าตัวประกอบกำลังไฟฟ้าและการเปลี่ยนแปลงของอุณหภูมิภายในห้อง ขณะใช้งานเครื่องปรับอากาศผ่านการสื่อสารแบบไร้สายเข้าสู่ระบบอินเทอร์เน็ต เพื่อนำค่าที่ได้ไปเก็บในฐานข้อมูลบนเครื่องแม่ข่าย รวมถึงสามารถนำข้อมูลที่บันทึกไว้มาแสดงผลในรูปแบบกราฟให้กับผู้ใช้งานผ่านทางคอมพิวเตอร์ได้ จากผลการทดลองสรุปได้ว่า

- 1) สามารถสร้างอุปกรณ์การวัดที่สามารถวัดค่ากระแสไฟฟ้า, แรงดันไฟฟ้า, อุณหภูมิ, กำลังไฟฟ้า, ค่าตัวประกอบกำลังไฟฟ้า, วันที่และเวลา โดยใช้โมดูลเซนเซอร์วัดกระแสไฟฟ้า (ACS712-20A), โมดูลเซนเซอร์วัดแรงดันไฟฟ้า (ZMPT101B), โมดูลเซนเซอร์วัดอุณหภูมิ (AM2302), โมดูลนาฬิกา (Real Time Clock DS3231) และส่งข้อมูลผ่านไมโครคอนโทรลเลอร์พร้อมระบบเครือข่ายแบบไร้สาย NodeMCU V3 เพื่อนำข้อมูลดังกล่าวไปเก็บที่ฐานข้อมูล MySQL โดยข้อมูลที่วัดได้จะถูกส่งเข้าไปเก็บในฐานข้อมูลทุก ๆ ระยะเวลา 1 วินาที ผู้ใช้งานสามารถดูข้อมูลในฐานข้อมูล MySQL ได้โดยการดูผ่านหน้าเว็บเซิร์ฟเวอร์ phpMyAdmin
- 2) ข้อมูลจากการวัดค่ากระแสไฟฟ้าและแรงดันไฟฟ้า นั้นมีความถูกต้องแม่นยำและมีความน่าเชื่อถือสูงเนื่องจากค่าที่วัดได้มีเปอร์เซ็นต์ความผิดพลาดเพียงเล็กน้อย เมื่อนำไปเปรียบเทียบกับอุปกรณ์มาตรฐานดิจิทัลแคลมป์มิเตอร์ ยี่ห้อ KYORITSU ข้อมูลกระแสไฟฟ้าจากอุปกรณ์วัดมีเปอร์เซ็นต์ความผิดพลาดเฉลี่ยเท่ากับ 1.78% และข้อมูลแรงดันไฟฟ้าจากอุปกรณ์วัดมีเปอร์เซ็นต์ความผิดพลาดเฉลี่ยเท่ากับ 0.17% จึงทำให้ค่ากำลังไฟฟ้าและค่าตัวประกอบกำลังไฟฟ้า ที่ได้จากการคำนวณค่ากระแสไฟฟ้าและแรงดันไฟฟ้านั้นมีความน่าเชื่อถือสูง
- 3) การนำข้อมูลจากฐานข้อมูล MySQL มาแสดงผลในรูปแบบกราฟตามช่วงเวลาที่ต้องการดูข้อมูลบนเว็บเพจที่สร้างขึ้นนั้น พบว่า เมื่อทำการเลือกช่วงวันที่ที่ต้องการให้แสดงข้อมูลแล้ว กราฟข้อมูลกระแสไฟฟ้า, แรงดันไฟฟ้า, อุณหภูมิ, กำลังไฟฟ้าและค่าตัวประกอบกำลังไฟฟ้าจะปรากฏบนหน้าเว็บเพจเมื่อเวลาผ่านไป 30 วินาที ถือว่ามีความล่าช้าผิดปกติอาจเกิดจากการที่ระบบต้องใช้เวลาในการสร้างภาพกราฟิก (Rendering) เพราะข้อมูลในฐานข้อมูลมีจำนวนมาก ซึ่งเกี่ยวข้องกับประสิทธิภาพของเครื่องคอมพิวเตอร์ที่ใช้ในการประมวลผลเช่นกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2 ข้อเสนอแนะ

- 1) เพื่อให้ระบบสามารถเก็บข้อมูลการวัดปริมาณกระแสไฟฟ้า, แรงดันไฟฟ้า, อุณหภูมิ, กำลังไฟฟ้า, ค่าตัวประกอบกำลังไฟฟ้า, วันที่และเวลา ในระยะเวลายาวนานควรใช้เครื่องคอมพิวเตอร์เซิร์ฟเวอร์แทนการจำลองคอมพิวเตอร์เป็นเว็บเซิร์ฟเวอร์ เพื่อกำจัดปัญหาขนาดหน่วยความจำและลดความล่าช้าในการทำงานของระบบ
- 2) ควรติดตั้งโมดูลการ์ดหน่วยความจำ (Micro SD Card) เพื่อสำรองข้อมูลในขณะที่ระบบเครือข่ายแบบไร้สายขัดข้องจนไม่สามารถส่งข้อมูล จากคอนโทรลเลอร์หลักไปสู่ฐานข้อมูลบนเครื่องแม่ข่ายได้ เมื่อระบบเครือข่ายไร้สายกลับมาใช้งานได้เป็นปกติก็จะสามารถส่งข้อมูลย้อนหลังได้ ทำให้ข้อมูลการวัดช่วงนั้นไม่เกิดหายไป



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

- [1] Ivan Grokhotkov, “ESP8266 Arduino Core Documentation : Release 2.4.0”, Shanghai : 2016.
- [2] ชาญชัย ศุภอรรถกร, “สร้างเว็บแอปพลิเคชัน PHP MySQL+AJAX jQuery ฉบับสมบูรณ์”, กรุงเทพฯ : ไร้ไวร์, 2560.
- [3] Luke Welling & Laura Thomson. “PHP and MySQL Web Development”, New Jersey, United States : 2016.
- [4] Berners-Lee, “HTML Hyper Text Markup Language”, Telangana : Tutorials Point, 2012.
- [5] Espressif IOT Team, “ESP8266EX Datasheet”, Feb 2018. [Online] Available : <https://www.espressif.com/products/hardware/esp8266ex/overview> (7 May 2018).
- [6] Microship Technology Inc, “MCP3004/MCP3008”, 2007. [Online] Available : <https://cdn-shop.adafruit.com/datasheets/MCP3008.pdf> (May 7, 2018).
- [7] Allegro Microsystem Inc., “ACS712-DS Rev. 7”, 2007. [Online] Available : <https://www.sparkfun.com/datasheets/BreakoutBoards/0712.pdf> (May 7, 2018).
- [8] Aosong(Guangzhou) Electronics Co.,Ltd., “Temperature and humidity module AM2302 Product Manual”, 2018. [Online] Available : <https://akizukidenshi.com/download/ds/aosong/AM2302.pdf> (May 7,2018).
- [9] Maxim Integrated, Mar 2015. [Online] Available : <https://datasheets.maximintegrated.com/en/ds/DS3231.pdf> (May 8, 2018).
- [10] สุกิจ คุชัยสิทธิ์, “ความรู้เบื้องต้นเกี่ยวกับระบบจัดการฐานข้อมูล ตอนที่ 1”, กรุงเทพฯ : สำนักพิมพ์มหาวิทยาลัยกรุงเทพ, 2561.
- [11] MarcusCode, “ภาษา C”, 27 มิถุนายน 2558. [ระบบออนไลน์] แหล่งที่มา : <http://marcuscode.com/lang/c> (May 23, 2018).
- [12] Mindphp, “PHP คืออะไร”, 14 มีนาคม 2560. [ระบบออนไลน์] แหล่งที่มา : <http://www.mindphp.com/คู่มือ/73-คืออะไร/2127-php-คืออะไร.html> (May 10, 2018).
- [13] นันทวัฒน์ ไชยรัตน์, “โครงสร้างของภาษา HTML”, 13 กันยายน 2560. [ระบบออนไลน์] แหล่งที่มา : https://www.hellomyweb.com/course/html/structure_of_html/ (May 10, 2018).
- [14] วิกิพีเดีย, “ไฟฟ้ากระแสสลับ”, 24 กุมภาพันธ์ 2561. [ระบบออนไลน์] แหล่งที่มา : <https://th.wikipedia.org/wiki/ไฟฟ้ากระแสสลับ> (May 11, 2018).

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรม ติดต่อฐานข้อมูล MySQL โดยใช้ภาษา PHP เขียนในโปรแกรม Notepad

```

<?php
$servername = "localhost";
$username = "root";
$password = "";
$dbname = "project1";
$conn = new mysqli($servername, $username,$password, $dbname);
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}
$val = array($_GET['AmpsRMS'],$_GET['Volt'],$_GET['temp'],$_GET['RealPower'],$_GET
['PowerFactor'],$_GET['date'],$_GET['time'],$_GET['datecheck'],$_GET['APOWER'],
$_GET['DM']);
$sql = "INSERT INTO
data(current,voltage,temp,REALPOWER,PF,date,time,datecheck,APPOWER,DM) VALUES
('$val[0]','$val[1]','$val[2]','$val[3]','$val[4]','$val[5]','$val[6]','$val[7]','$val[8]','$val[9]')";

if ($conn->query($sql) === TRUE) {
    echo "Save OK";
} else
{
    echo "Error:" . $sql . "<br>" . $conn->error;
}
$conn->close();
?>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรม รับค่าจากโมดูลเซนเซอร์แล้วส่งข้อมูลจาก NodeMCU ไปเก็บในฐานข้อมูล MySQL เขียนบน
โปรแกรม Arduino IDE

```
#include "EmonLib.h" // Include Emon Library
EnergyMonitor emon1; // Create an instance
#include <b64.h>
#include <HttpClient.h>
//-----
#include "DHT.h"
#define DHTPIN D3
#define DHTTYPE DHT22
DHT dht(DHTPIN, DHTTYPE);
//-----
#include <Arduino.h>
#include <ESP8266WiFi.h>
#include <ESP8266WiFiMulti.h>
#include <ESP8266HTTPClient.h>
ESP8266WiFiMulti WiFiMulti;
//-----
#include <Wire.h> //I2C library
#include <RtcDS3231.h> //RTC library
RtcDS3231<TwoWire> rtcObject(Wire);
float VA ;
float PF ;
float Volt ;
float AmpsRMS ;
float t ;
void setup()
{
Serial.begin(115200);
rtcObject.Begin(); //Starts I2C
delay(100);
RtcDateTime currentTime = RtcDateTime(18, 4, 23,18 ,0 ,20); //ตั้งค่าวันที่และเวลา
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


```

rtcObject.SetDateTime(currentTime);
WiFiMulti.addAP("POY", "0903380339"); // ssid, password
emon1.voltage(0, 500,-5.5);
emon1.current(2, 15.6);
}
void loop() {
if ((WiFiMulti.run() == WL_CONNECTED)) {
emon1.calcVI(50,2000);
WATT = abs(emon1.realPower); //extract Real Power into variable
VA = abs(emon1.apparentPower); //extract Apparent Power into variable
PF = (emon1.powerFactor); //extract Power Factor into Variable
Volt = abs(emon1.Vrms); //extract Vrms into Variable
AmpsRMS = emon1.Irms;
if(Volt <10)
{ VA=0;
WATT = 0;
PF=0;
AmpsRMS =0;
}
/* Serial.print(AmpsRMS);
Serial.println(" Amps RMS");
Serial.print(Volt);
Serial.println(" Volt rms ");
Serial.print(Volt*AmpsRMS);
Serial.println(" VA ");*/
//----- Read Temp and Time -----
t = dht.readTemperature();
RtcDateTime currentTime = rtcObject.GetDateTime(); //get the time from the RTC
char str1[10];
char str2[8];
char str3[4];
sprintf(str1, "%d:%d:%d",currentTime.Year(),currentTime.Month(),currentTime.Day() );

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

sprintf(str2, "%d:%d:%d", currentTime.Hour(), currentTime.Minute(),currentTime.Second() );
sprintf(str3, "%d%d",currentTime.Month(),currentTime.Day() );
int DAY = currentTime.Day();
//-----Sent Data -----
while(t==0)
{
delay(200);
t = dht.readTemperature();
if (t=0)
{delay(200);
t = dht.readTemperature();}
}
HTTPClient http;
String url =
"http://192.168.1.102/project/add.php?AmpsRMS="+String(AmpsRMS,2)+"&Volt="+String(Volt)+
"&temp="+String(t,1)+"&RealPower="+String(WATT,1)+"&APOWER="+String(VA,1)+"&PowerFacto
r="+String(PF,2)+"&date="+String(str1)+"&time="+String(str2)+"&datecheck="+String(DAY)+"&DM
="+String(str3) ;
Serial.println("Sending Data To MySQL Server ");
http.begin(url); //HTTP
delay(1000);
int httpCode = http.GET();
if (httpCode > 0) {
Serial.printf("[HTTP] GET... code: %d\n", httpCode);
if (httpCode == HTTP_CODE_OK) {
String payload = http.getString();
Serial.println(payload);
}
} else {Serial.printf("[HTTP] GET... failed, error: %s\n", http.errorToString(httpCode).c_str());}
http.end();
delay(5000);
}}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรม สร้างเว็บแสดงผลบนโปรแกรม Notepad++

```

<?php
ini_set('memory_limit', '1024M');
$connect = mysqli_connect("localhost", "root", "", "project1");
$t1 = $_GET["Start"];
$t2 = $_GET["End"];
//$t4 = mysqli_real_escape_string($connect,$t3);

$query= 'SELECT Voltage, UNIX_TIMESTAMP(CONCAT_WS(" ", date,time)) AS datetime FROM
data Where DM between '.$t1.' AND '.$t2.' ORDER BY date DESC, time DESC';
//$query= 'SELECT Voltage, UNIX_TIMESTAMP(CONCAT_WS(" ", date,time)) AS datetime FROM
data WHERE date= '.$t3.' AND date = '.$t4.' ORDER BY date DESC, time DESC';

$result = mysqli_query($connect, $query);
$rows = array();
$table = array();

$query2 = 'SELECT Current, UNIX_TIMESTAMP(CONCAT_WS(" ", date, time)) AS datetime
FROM data Where DM between '.$t1.' AND '.$t2.' ORDER BY date DESC, time DESC' ;
$result2 = mysqli_query($connect, $query2);
$rows2 = array();
$table2 = array();

$query3 = 'SELECT Temp, UNIX_TIMESTAMP(CONCAT_WS(" ", date, time)) AS datetime FROM
data Where DM between '.$t1.' AND '.$t2.' ORDER BY date DESC, time DESC' ;
$result3 = mysqli_query($connect, $query3);
$rows3 = array();
$table3 = array();

$query4 = 'SELECT REALPOWER, APPPOWER, UNIX_TIMESTAMP(CONCAT_WS(" ", date, time)) AS
datetime FROM data Where DM between '.$t1.' AND '.$t2.' ORDER BY date DESC, time DESC' ;
$result4 = mysqli_query($connect, $query4);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

$rows4 = array();
$table4 = array();

$query5 = 'SELECT PF, UNIX_TIMESTAMP(CONCAT_WS(" ", date, time)) AS datetime FROM
data Where DM between '.$t1.' AND '.$t2.' ORDER BY date DESC, time DESC' ;
$result5 = mysqli_query($connect, $query5);
$rows5 = array();
$table5 = array();

$table['cols'] = array(
array(
'label' => 'Date Time',
'type' => 'datetime'
),
array(
'label' => 'Voltage (V)',
'type' => 'number'
)
);
while($row = mysqli_fetch_array($result))
{
$sub_array = array();
$date_time = explode(".", $row["datetime"]);
$sub_array[] = array(
"v" => 'Date(' . $date_time[0] . '000'
);
$sub_array[] = array(
"v" => $row["Voltage"]
);
$rows[] = array(
"v" => $sub_array
);
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

}
$table['rows'] = $rows;
$jsonTable = json_encode($table);

$table2['cols'] = array(
    array(
        'label' => 'Date Time',
        'type' => 'datetime'
    ),
    array(
        'label' => 'Current (A)',
        'type' => 'number'
    )
);
while($row2 = mysqli_fetch_array($result2))
{
    $sub_array2 = array();
    $datetime2 = explode(".", $row2["datetime"]);
    $sub_array2[] = array(
        "v" => 'Date(' . $datetime2[0] . '000)'
    );
    $sub_array2[] = array(
        "v" => $row2["Current"]
    );
    $rows2[] = array(
        "c" => $sub_array2
    );
}
$table2['rows'] = $rows2;
$jsonTable2 = json_encode($table2);

$table3['cols'] = array(

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

array(
  'label' => 'Date Time',
  'type' => 'datetime'
),
array(
  'label' => 'Temperature (°C)',
  'type' => 'number'
)
);
while($row3 = mysqli_fetch_array($result3))
{
  $sub_array3 = array();
  $datetime3 = explode(".", $row3["datetime"]);
  $sub_array3[] = array(
    "v" => 'Date(' . $datetime3[0] . '000)'
  );
  $sub_array3[] = array(
    "v" => $row3["Temp"]
  );
  $rows3[] = array(
    "c" => $sub_array3
  );
}
$table3['rows'] = $rows3;
$jsonTable3 = json_encode($table3);

$table4['cols'] = array(
  array(
    'label' => 'Date Time',
    'type' => 'datetime'
  ),
  array(

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

'label' => 'Real Power (Watt)',
'type' => 'number'
),
array(
'label' => 'Apparent Power (VA)',
'type' => 'number'
)
);
while($row4 = mysqli_fetch_array($result4))
{
$sub_array4 = array();
$datetime4 = explode(".", $row4["datetime"]);
$sub_array4[] = array(
    "v" => 'Date(' . $datetime4[0] . ', '000)'
);
$sub_array4[] = array(
    "v" => $row4["REALPOWER"]
);
$sub_array4[] = array(
    "v" => $row4["APPOWER"]
);
$rows4[] = array(
    "c" => $sub_array4
);
}
$table4['rows'] = $rows4;
$jsonTable4 = json_encode($table4);

$table5['cols'] = array(
array(
'label' => 'Date Time',
'type' => 'datetime'

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

),
array(
    'label' => 'Power Factor',
    'type' => 'number'
)
);
while($row5 = mysqli_fetch_array($result5))
{
    $sub_array5 = array();
    $datetime5 = explode(".", $row5["datetime"]);
    $sub_array5[] = array(
        "v" => 'Date(' . $datetime5[0] . '000)'
    );
    $sub_array5[] = array(
        "v" => $row5["PF"]
    );
    $rows5[] = array(
        "c" => $sub_array5
    );
}
$table5['rows'] = $rows5;
$jsonTable5 = json_encode($table5);
?>

<html>
<head>
<script src="https://code.jquery.com/jquery-1.12.4.js"></script>
<script src="https://cdnjs.cloudflare.com/ajax/libs/jquery-scrollTo/2.1.2/jquery.scrollTo.min.js"></script>
<script src="https://cdnjs.cloudflare.com/ajax/libs/jquery-localScroll/2.0.0/jquery.localScroll.min.js"></script>
<script src="https://cdnjs.cloudflare.com/ajax/libs/smooth-scroll/12.1.5/js/smooth-

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


```

scroll.min.js"></script>
<script type="text/javascript" src="https://www.gstatic.com/charts/loader.js"></script>
<script type="text/javascript"
src="//ajax.googleapis.com/ajax/libs/jquery/1.10.2/jquery.min.js"></script>
<link rel="stylesheet" href="//code.jquery.com/ui/1.12.1/themes/base/jquery-ui.css">
<link rel="stylesheet" href="/resources/demos/style.css">
<script src="https://code.jquery.com/jquery-1.12.4.js"></script>
<script src="https://code.jquery.com/ui/1.12.1/jquery-ui.js"></script>
<script>
  google.charts.load('current', {'packages':['corechart']});
  google.charts.setOnLoadCallback(drawChart1);
  google.charts.setOnLoadCallback(drawChart2);
  google.charts.setOnLoadCallback(drawChart3);
  google.charts.setOnLoadCallback(drawChart4);
  google.charts.setOnLoadCallback(drawChart5);

function drawChart1()
{
  var data = new google.visualization.DataTable(<?php echo $jsonTable; ?>);
  var options = {
    title:'Voltage',
    legend:{position:'bottom'},
    vAxis: {minValue: 0},
    chartArea:{width:'90%', height: '65%'},
    animation:{
      duration: 1000,
      easing: 'linear',
      startup: true
    },
    explorer: {
      actions: ['dragToZoom', 'rightClickToReset'],
      axis: 'horizontal',

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        keepInBounds: true,
        maxZoomIn: 8.0
    }
};

var chart1 = new
google.visualization.LineChart(document.getElementById('line_chart'));
chart1.draw(data, options);
}

function drawChart2()
{
var data2 = new google.visualization.DataTable(<?php echo $jsonTable2; ?>);
var options2 = {
title:'Current',
legend:{position:'bottom'},
vAxis: {minValue: 0},
chartArea:{width:'90%', height: '65%'},
animation:{
duration: 1000,
easing: 'linear',
startup: true
},

explorer: {
actions: ['dragToZoom', 'rightClickToReset'],
axis: 'horizontal',
keepInBounds: true,
maxZoomIn: 8.0
},

colors:['red','#004411']
};

var chart2 = new
google.visualization.LineChart(document.getElementById('line_chart2'));

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

chart2.draw(data2, options2);
}

function drawChart3()
{
var data3 = new google.visualization.DataTable(<?php echo $jsonTable3; ?>);

var options3 = {
title:'Temperature',
legend:{position:'bottom'},
vAxis: {minValue: 0},
chartArea:{width:'90%', height: '65%' },
animation:{
duration: 1000,
easing: 'linear',
startup: true
},
explorer: {
actions: ['dragToZoom', 'rightClickToReset'],
axis: 'horizontal',
keepInBounds: true,
maxZoomIn: 8.0
},
colors:['green']
};

var chart3 = new
google.visualization.LineChart(document.getElementById('line_chart3'));
chart3.draw(data3, options3);
}

function drawChart4()
{

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

var data4 = new google.visualization.DataTable(<?php echo $jsonTable4; ?>);

var options4 = {
  title:'Power',
  legend:{position:'bottom'},
  vAxis: {minValue: 0},
  chartArea:{width:'90%', height: '65%' },
  animation:{
    duration: 1000,
    easing: 'linear',
    startup: true
  },
  explorer: {
    actions: ['dragToZoom', 'rightClickToReset'],
    axis: 'horizontal',
    keepInBounds: true,
    maxZoomIn: 8.0
  },
  colors:['purple','orange']
};

var chart4 = new
google.visualization.LineChart(document.getElementById("line_chart4"));
chart4.draw(data4, options4);
}

function drawChart5()
{
  var data5 = new google.visualization.DataTable(<?php echo $jsonTable5; ?>);

  var options5 = {
    title:'Power Factor',
    legend:{position:'bottom'},

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    vAxis: {minValue: 0},
    chartArea:{width:'90%', height: '65%' },
    animation:{
        duration: 1000,
        easing: 'linear',
        startup: true
    },
    explorer: {
        actions: ['dragToZoom', 'rightClickToReset'],
        axis: 'horizontal',
        keepInBounds: true,
        maxZoomIn: 8.0
    },
    colors:['yellow']
};
var chart5 = new
google.visualization.LineChart(document.getElementById('line_chart5'));
chart5.draw(data5, options5);
}

$( function() {
    from = $( "#from" ).datepicker({
        dateFormat: "yy-mm-dd",
        defaultDate: "+1w",
        changeMonth: true,
        numberOfMonths: 2
    })
    .on( "change", function() {
        to.datepicker( "option", "minDate", getDate( this ) );
    }
    ),
    to = $( "#to" ).datepicker({
        dateFormat: "yy-mm-dd",

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        defaultDate: "+1w",
        changeMonth: true,
        numberOfMonths: 2
    })
    .on( "change", function() {
        from.datepicker( "option", "maxDate", getDate( this ) );
    });

    function getDate( element ) {
    var date;
    try {
        date = $.datepicker.parseDate( dateFormat, element.value );
    } catch( error ) {
        date = null;
    }
    return date;
    }
    });
</script>

<script src="jquery.min.js"></script>
<script>
$(function() {
$('a[href*=#]:not([href=#])').click(function() {
    var target = $(this.hash);
    target = target.length ? target : $('[name=' + this.hash.substr(1) +']');
    if (target.length) {
        $('html,body').animate({
            scrollTop: target.offset().top
        }, 1000);
        return false;
    }
}
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

});
});
</script>

<style>
.page-wrapper
{
width:1000px;
margin:0 auto;
}
</style>
<title>Project</title>
</head>

<body>
<div class="page-wrapper">
<br />
<h2 align="center">THE CURRENT LOGGING SYSTEM WITH WIRELESS LAN</h2>
<h3 align="center">EXAMPLE : Air Conditionner Sharp 12,000 BTU/Hr with 21.5 sq.m room
size </h3>
<p align="center"><script>document.write(Date());</script></p>
<form action="test.php" method="get" align="center">
<label for="from">Start Date</label>
<input type="text" id="Start" name="Start" placeholder = "04282018">
<label for="from">To</label>
<input type="text" id="End" name="End" placeholder = "04282018">
<input type="submit" value ="Submit" /> <br />
</form>

<div id="Exam" align="center">
<label > EXAMPLE FOR FILLING DATE : APRIL 22 2018 FILL AS 04222018
</label>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

</div>

<div id="scrollTo" align="center">
<a href="#line_chart2">Current</a> :
<a href="#line_chart">Voltage</a> :
<a href="#line_chart3">Temperature</a> :
<a href="#line_chart4">Power</a> :
<a href="#line_chart5">Power Factor</a>
</div>

```

```

<div id="line_chart2" style="width: 100%; height: 500px"></div>
<div id="line_chart" style="width: 100%; height: 500px"></div>
<div id="line_chart3" style="width: 100%; height: 500px"></div>
<div id="line_chart4" style="width: 100%; height: 500px"></div>
<div id="line_chart5" style="width: 100%; height: 500px"></div>

```

```

</div>
</body>
</html>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรม วัดค่ากระแสไฟฟ้า แรงดันไฟฟ้า กำลังไฟฟ้าและตัวประกอบกำลังไฟฟ้าในไลบรารี EmonLib.h

```
#include "EmonLib.h"
#ifdef ARDUINO && ARDUINO >= 100
#include "Arduino.h"
#else
#include "WProgram.h"
#endif
// Sets the pins to be used for voltage and current sensors
void EnergyMonitor::voltage(unsigned int _inPinV, double _VCAL, double _PHASECAL)
{
inPinV = _inPinV;
VCAL = _VCAL;
PHASECAL = _PHASECAL;
offsetV = ADC_COUNTS>>1;
}
void EnergyMonitor::current(unsigned int _inPinI, double _ICAL)
{
inPinI = _inPinI;
ICAL = _ICAL;
offsetI = ADC_COUNTS>>1;
}
// Sets the pins to be used for voltage and current sensors based on emontx pin map
void EnergyMonitor::voltageTX(double _VCAL, double _PHASECAL)
{
inPinV = 2;
VCAL = _VCAL;
PHASECAL = _PHASECAL;
offsetV = ADC_COUNTS>>1;
}
void EnergyMonitor::currentTX(unsigned int _channel, double _ICAL)
{
if (_channel == 1) inPinI = 3;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if (_channel == 2) inPinI = 0;
if (_channel == 3) inPinI = 1;
ICAL = _ICAL;
offsetI = ADC_COUNTS>>1;
}
// emon_calc procedure
// Calculates realPower,apparentPower,powerFactor,Vrms,Irms,kWh increment
// From a sample window of the mains AC voltage and current.
// The Sample window length is defined by the number of half wavelengths or crossings we
choose to measure.
void EnergyMonitor::calcVI(unsigned int crossings, unsigned int timeout)
{
#ifdef emonTxV3
int SupplyVoltage=3300;
#else
int SupplyVoltage = readVcc();
#endif

unsigned int crossCount = 0; //Used to measure number of times threshold is crossed.
unsigned int numberOfSamples = 0; //This is now incremented

// 1) Waits for the waveform to be close to 'zero' (mid-scale adc) part in sin curve.
boolean st=false; //an indicator to exit the while loop
unsigned long start = millis(); //millis()-start makes sure it doesnt get stuck in the loop if
there is an error.
while(st==false) //the while loop...
{
startV = adc.readADC(inPinV); //using the voltage waveform
if ((startV < (ADC_COUNTS*0.55)) && (startV > (ADC_COUNTS*0.45))) st=true; //check its within
range
if ((millis()-start)>timeout) st = true;
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

// 2) Main measurement loop
start = millis();
while ((crossCount < crossings) && ((millis()-start)<timeout))
{
numberOfSamples++; //Count number of times looped.
lastFilteredV = filteredV; //Used for delay/phase compensation
// A) Read in raw voltage and current samples
sampleV = adc.readADC(inPinV); //Read in raw voltage signal
sampleI = adc.readADC(inPinI); //Read in raw current signal
// B) Apply digital low pass filters to extract the 2.5 V or 1.65 V dc offset,
// then subtract this - signal is now centred on 0 counts.
offsetV = offsetV + ((sampleV-offsetV)/1024);
filteredV = sampleV - offsetV;
offsetI = offsetI + ((sampleI-offsetI)/1024);
filteredI = sampleI - offsetI;
// C) Root-mean-square method voltage
sqV= filteredV * filteredV; //1) square voltage values
sumV += sqV; //2) sum
// D) Root-mean-square method current
sqI = filteredI * filteredI; //1) square current values
sumI += sqI; //2) sum
// E) Phase calibration
phaseShiftedV = lastFilteredV + PHASECAL * (filteredV - lastFilteredV);
// F) Instantaneous power calc
instP = phaseShiftedV * filteredI; //Instantaneous Power
sumP +=instP; //Sum
// G) Find the number of times the voltage has crossed the initial voltage
// - every 2 crosses we will have sampled 1 wavelength
// - so this method allows us to sample an integer number of half wavelengths which
increases accuracy
lastVCross = checkVCross;
if (sampleV > startV) checkVCross = true;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

else checkVCross = false;
if (numberOfSamples==1) lastVCross = checkVCross;
if (lastVCross != checkVCross) crossCount++;
}
// 3) Post loop calculations
//Calculation of the root of the mean of the voltage and current squared (rms)
//Calibration coefficients applied.
double V_RATIO = VCAL *((SupplyVoltage/1000.0) / (ADC_COUNTS));
Vrms = V_RATIO * sqrt(sumV / numberOfSamples);
double I_RATIO = ICAL *((SupplyVoltage/1000.0) / (ADC_COUNTS));
Irms = I_RATIO * sqrt(sumI / numberOfSamples);
//Calculation power values
realPower = V_RATIO * I_RATIO * sumP / numberOfSamples;
apparentPower = Vrms * Irms;
powerFactor=realPower / apparentPower;
//Reset accumulators
sumV = 0;
sumI = 0;
sumP = 0;
}
double EnergyMonitor::calcIrms(unsigned int Number_of_Samples)
{
#ifdef emonTxV3
int SupplyVoltage=3300;
#else
int SupplyVoltage = readVcc();
#endif
for (unsigned int n = 0; n < Number_of_Samples; n++)
{
sampleI = adc.readADC(inPinI);
// Digital low pass filter extracts the 2.5 V or 1.65 V dc offset,
// then subtract this - signal is now centered on 0 counts.

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

offsetl = (offsetl + (samplel-offsetl)/1024);
filteredl = samplel - offsetl;
// Root-mean-square method current
// 1) square current values
sql = filteredl * filteredl;
// 2) sum
suml += sql;
}
double I_RATIO = ICAL * ((SupplyVoltage/1000.0) / (ADC_COUNTS));
Irms = I_RATIO * sqrt(suml / Number_of_Samples);
//Reset accumulators
suml = 0;

return Irms;
}
void EnergyMonitor::serialprint()
{
Serial.print(realPower);
Serial.print(' ');
Serial.print(apparentPower);
Serial.print(' ');
Serial.print(Vrms);
Serial.print(' ');
Serial.print(Irms);
Serial.print(' ');
Serial.print(powerFactor);
Serial.println(' ');
delay(100);
}

long EnergyMonitor::readVcc() {
long result;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

#if defined(_AVR_ATmega168_) || defined(_AVR_ATmega328_) || defined (_AVR_ATmega328P_)
ADMUX = _BV(REFS0) | _BV(MUX3) | _BV(MUX2) | _BV(MUX1);
#elif defined(_AVR_ATmega644_) || defined(_AVR_ATmega644P_) ||
defined(_AVR_ATmega1284_) || defined(_AVR_ATmega1284P_)
ADMUX = _BV(REFS0) | _BV(MUX4) | _BV(MUX3) | _BV(MUX2) | _BV(MUX1);
#elif defined(_AVR_ATmega32U4_) || defined(_AVR_ATmega1280_) ||
defined(_AVR_ATmega2560_) || defined(_AVR_AT90USB1286_)
ADMUX = _BV(REFS0) | _BV(MUX4) | _BV(MUX3) | _BV(MUX2) | _BV(MUX1);
ADCSRB &= ~_BV(MUX5); // Without this the function always returns -1 on the ATmega2560
http://openenergymonitor.org/emon/node/2253#comment-11432
#elif defined (_AVR_ATtiny24_) || defined(_AVR_ATtiny44_) || defined(_AVR_ATtiny84_)
ADMUX = _BV(MUX5) | _BV(MUX0);
#elif defined (_AVR_ATtiny25_) || defined(_AVR_ATtiny45_) || defined(_AVR_ATtiny85_)
ADMUX = _BV(MUX3) | _BV(MUX2);
#endif

#if defined(_AVR_)
delay(2); // Wait for Vref to settle
ADCSRA |= _BV(ADSC); // Convert
while (bit_is_set(ADCSRA,ADSC));
result = ADCL;
result |= ADCH<<8;
result = READVCC_CALIBRATION_CONST / result; //1100mV*1024 ADC steps
http://openenergymonitor.org/emon/node/1186
return result;
#elif defined(_arm_)
return (3300); //Arduino Due
#else
return (3300); //Guess that other un-supported architectures will be running a 3.3V!
#endif
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



1.

Overview

Espressif's ESP8266EX delivers highly integrated Wi-Fi SoC solution to meet users' continuous demands for efficient power usage, compact design and reliable performance in the Internet of Things industry.

With the complete and self-contained Wi-Fi networking capabilities, ESP8266EX can perform either as a standalone application or as the slave to a host MCU. When ESP8266EX hosts the application, it promptly boots up from the flash. The integrated high-speed cache helps to increase the system performance and optimize the system memory. Also, ESP8266EX can be applied to any microcontroller design as a Wi-Fi adaptor through SPI / SDIO or I2C / UART interfaces.

ESP8266EX integrates antenna switches, RF balun, power amplifier, low noise receive amplifier, filters and power management modules. The compact design minimizes the PCB size and requires minimal external circuitries.

Besides the Wi-Fi functionalities, ESP8266EX also integrates an enhanced version of Tensilica's L106 Diamond series 32-bit processor and on-chip SRAM. It can be interfaced with external sensors and other devices through the GPIOs. Software Development Kit (SDK) provides sample codes for various applications.

Espressif Systems' Smart Connectivity Platform (ESCP) enables sophisticated features including fast switch between sleep and wakeup mode for energy-efficient purpose, adaptive radio biasing for low-power operation, advance signal processing, spur cancellation and radio co-existence mechanisms for common cellular, Bluetooth, DDR, LVDS, LCD interference mitigation.

1.1. Wi-Fi Protocols

- 802.11 b/g/n support
- 2 x Wi-Fi interface, supports infrastructure BSS Station mode / P2P mode / SoftAP mode support
- Hardware accelerators for CCMP (CBC-MAC, counter mode), TKIP (MIC, RC4), WAPI (SMS4), WEP (RC4), CRC
- 802.11n support (2.4 GHz)
- Supports MIMO 1x1 and 2x1, STBC, and 0.4 μ s guard interval
- WMM
- UMA compliant and certified
- Antenna diversity and selection (software managed hardware)
- Configurable packet traffic arbitration (PTA) with dedicated slave processor based design provides flexible and exact timing Bluetooth co-existence support for a wide range of Bluetooth Chip vendor.



- Dual and single antenna Bluetooth co-existence support with optional simultaneous receive (Wi-Fi/Bluetooth) capability

1.2. Specifications

Table 1-1. Specifications

Categories	Items	Parameters
Wi-Fi	Certification	Wi-Fi Alliance
	Protocols	802.11 b/g/n
	Frequency Range	2.4G ~ 2.5G (2400M ~ 2483.5M)
	Tx Power	802.11 b: +20 dBm
		802.11 g: +17 dBm
		802.11 n: +14 dBm
	Rx Sensitivity	802.11 b: -91 dbm (11 Mbps)
		802.11 g: -75 dbm (54 Mbps)
		802.11 n: -72 dbm (MCS7)
	Antenna	PCB Trace, External, IPEX Connector, Ceramic Chip
CPU	Tensilica L106 32-bit processor	
Hardware	Peripheral Interface	UART/SDIO/SPI/I2C/I2S/IR Remote Control
		GPIO/ADC/PWM/LED Light & Button
	Operating Voltage	2.5V ~ 3.6V
	Operating Current	Average value: 80 mA
	Operating Temperature Range	-40°C ~ 125°C
	Storage Temperature Range	-40°C ~ 125°C
	Package Size	QFN32-pin (5 mm x 5 mm)
	External Interface	-
	Wi-Fi Mode	Station/SoftAP/SoftAP+Station
	Security	WPA/WPA2
Software	Encryption	WEP/TKIP/AES
	Firmware Upgrade	UART Download / OTA (via network)
	Software Development	Supports Cloud Server Development / Firmware and SDK for fast on-chip programming
	Network Protocols	IPv4, TCP/UDP/HTTP/FTP
	User Configuration	AT Instruction Set, Cloud Server, Android/iOS App



1.3. Applications

- Home appliances
- Home automation
- Smart plugs and lights
- Mesh network
- Industrial wireless control
- Baby monitors
- IP cameras
- Sensor networks
- Wearable electronics
- Wi-Fi location-aware devices
- Security ID tags
- Wi-Fi position system beacons





MCP3004/3008

2.7V 4-Channel/8-Channel 10-Bit A/D Converters with SPI™ Serial Interface

Features

- 10-bit resolution
- ± 1 LSB max DNL
- ± 1 LSB max INL
- 4 (MCP3004) or 8 (MCP3008) input channels
- Analog inputs programmable as single-ended or pseudo-differential pairs
- On-chip sample and hold
- SPI serial interface (modes 0,0 and 1,1)
- Single supply operation: 2.7V - 5.5V
- 200 ksp/s max. sampling rate at $V_{DD} = 5V$
- 75 ksp/s max. sampling rate at $V_{DD} = 2.7V$
- Low power CMOS technology
- 5 nA typical standby current, 2 μA max.
- 500 μA max. active current at 5V
- Industrial temp range: $-40^{\circ}C$ to $+85^{\circ}C$
- Available in PDIP, SOIC and TSSOP packages

Applications

- Sensor Interface
- Process Control
- Data Acquisition
- Battery Operated Systems

Package Types

PDIP, SOIC, TSSOP

CH0	1	14	V_{DD}
CH1	2	13	V_{REF}
CH2	3	12	AGND
CH3	4	11	CLK
NC	5	10	D _{OUT}
NC	6	9	D _{IN}
DGND	7	8	CS/SHDN

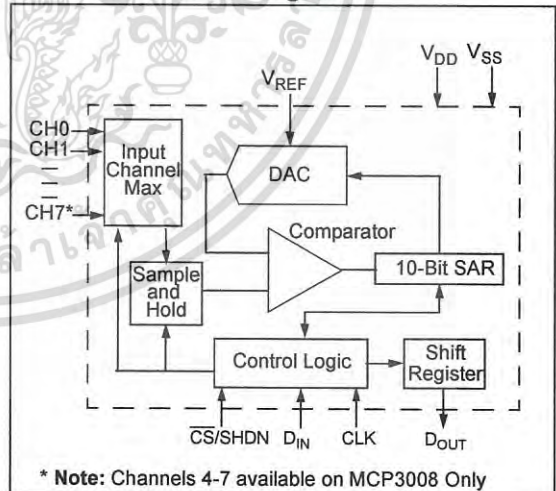
PDIP, SOIC

CH0	1	16	V_{DD}
CH1	2	15	V_{REF}
CH2	3	14	AGND
CH3	4	13	CLK
CH4	5	12	D _{OUT}
CH5	6	11	D _{IN}
CH6	7	10	CS/SHDN
CH7	8	9	DGND

Description

The Microchip Technology Inc. MCP3004/3008 devices are successive approximation 10-bit Analog-to-Digital (A/D) converters with on-board sample and hold circuitry. The MCP3004 is programmable to provide two pseudo-differential input pairs or four single-ended inputs. The MCP3008 is programmable to provide four pseudo-differential input pairs or eight single-ended inputs. Differential Nonlinearity (DNL) and Integral Nonlinearity (INL) are specified at ± 1 LSB. Communication with the devices is accomplished using a simple serial interface compatible with the SPI protocol. The devices are capable of conversion rates of up to 200 ksp/s. The MCP3004/3008 devices operate over a broad voltage range (2.7V - 5.5V). Low current design permits operation with typical standby currents of only 5 nA and typical active currents of 320 μA . The MCP3004 is offered in 14-pin PDIP, 150 mil SOIC and TSSOP packages, while the MCP3008 is offered in 16-pin PDIP and SOIC packages.

Functional Block Diagram



MCP3004/3008

1.0 ELECTRICAL CHARACTERISTICS

Absolute Maximum Ratings*

V_{DD} 7.0V
 All inputs and outputs w.r.t. V_{SS} -0.6V to $V_{DD} + 0.6V$
 Storage temperature -65°C to +150°C
 Ambient temp. with power applied -65°C to +125°C
 Soldering temperature of leads (10 seconds) .. +300°C
 ESD protection on all pins > 4 kV

***Notice:** Stresses above those listed under "Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at those or any other conditions above those indicated in the operation listings of this specification is not implied. Exposure to maximum rating conditions for extended periods may affect device reliability.

PIN FUNCTION TABLE

Name	Function
V_{DD}	+2.7V to 5.5V Power Supply
DGND	Digital Ground
AGND	Analog Ground
CH0-CH7	Analog Inputs
CLK	Serial Clock
D_{IN}	Serial Data In
D_{OUT}	Serial Data Out
$\overline{CS}/SHDN$	Chip Select/Shutdown Input
V_{REF}	Reference Voltage Input

ELECTRICAL SPECIFICATIONS

Electrical Characteristics: Unless otherwise noted, all parameters apply at $V_{DD} = 5V$, $V_{REF} = 5V$, $T_{AMB} = -40^{\circ}C$ to $+85^{\circ}C$, $f_{SAMPLE} = 200$ ksp/s and $f_{CLK} = 18 * f_{SAMPLE}$. Unless otherwise noted, typical values apply for $V_{DD} = 5V$, $T_{AMB} = 25^{\circ}C$.

Parameter	Sym	Min	Typ	Max	Units	Conditions
Conversion Rate						
Conversion Time	t_{CONV}	—	—	10	clock cycles	
Analog Input Sample Time	t_{SAMPLE}	—	1.5	—	clock cycles	
Throughput Rate	f_{SAMPLE}	—	—	200 75	ksp/s ksp/s	$V_{DD} = V_{REF} = 5V$ $V_{DD} = V_{REF} = 2.7V$
DC Accuracy						
Resolution		—	10	—	bits	
Integral Nonlinearity	INL	—	± 0.5	± 1	LSB	
Differential Nonlinearity	DNL	—	± 0.25	± 1	LSB	No missing codes over temperature
Offset Error		—	—	± 1.5	LSB	
Gain Error		—	—	± 1.0	LSB	
Dynamic Performance						
Total Harmonic Distortion		—	-76	—	dB	$V_{IN} = 0.1V$ to $4.9V @ 1$ kHz
Signal to Noise and Distortion (SINAD)		—	61	—	dB	$V_{IN} = 0.1V$ to $4.9V @ 1$ kHz
Spurious Free Dynamic Range		—	78	—	dB	$V_{IN} = 0.1V$ to $4.9V @ 1$ kHz
Reference Input						
Voltage Range		0.25	—	V_{DD}	V	Note 2
Current Drain		—	100 0.001	150 3	μA μA	$\overline{CS} = V_{DD} = 5V$

- Note 1:** This parameter is established by characterization and not 100% tested.
Note 2: See graphs that relate linearity performance to V_{REF} levels.
Note 3: Because the sample cap will eventually lose charge, effective clock rates below 10 kHz can affect linearity performance, especially at elevated temperatures. See Section 6.2, "Maintaining Minimum Clock Speed", for more information.

ELECTRICAL SPECIFICATIONS (CONTINUED)

Electrical Characteristics: Unless otherwise noted, all parameters apply at $V_{DD} = 5V$, $V_{REF} = 5V$, $T_{AMB} = -40^{\circ}C$ to $+85^{\circ}C$, $f_{SAMPLE} = 200$ ksp/s and $f_{CLK} = 18 \cdot f_{SAMPLE}$. Unless otherwise noted, typical values apply for $V_{DD} = 5V$, $T_{AMB} = 25^{\circ}C$.

Parameter	Sym	Min	Typ	Max	Units	Conditions	
Analog Inputs							
Input Voltage Range for CH0 or CH1 in Single-Ended Mode		V_{SS}	—	V_{REF}	V		
Input Voltage Range for IN+ in pseudo-differential mode		IN-	—	$V_{REF} + IN-$			
Input Voltage Range for IN- in pseudo-differential mode		$V_{SS} - 100$	—	$V_{SS} + 100$	mV		
Leakage Current		—	0.001	± 1	μA		
Switch Resistance		—	1000	—	Ω	See Figure 4-1	
Sample Capacitor		—	20	—	pF	See Figure 4-1	
Digital Input/Output							
Data Coding Format		Straight Binary					
High Level Input Voltage	V_{IH}	$0.7 V_{DD}$	—	—	V		
Low Level Input Voltage	V_{IL}	—	—	$0.3 V_{DD}$	V		
High Level Output Voltage	V_{OH}	4.1	—	—	V	$I_{OH} = -1$ mA, $V_{DD} = 4.5V$	
Low Level Output Voltage	V_{OL}	—	—	0.4	V	$I_{OL} = 1$ mA, $V_{DD} = 4.5V$	
Input Leakage Current	I_{LI}	-10	—	10	μA	$V_{IN} = V_{SS}$ or V_{DD}	
Output Leakage Current	I_{LO}	-10	—	10	μA	$V_{OUT} = V_{SS}$ or V_{DD}	
Pin Capacitance (All Inputs/Outputs)	C_{IN} , C_{OUT}	—	—	10	pF	$V_{DD} = 5.0V$ (Note 1) $T_{AMB} = 25^{\circ}C$, $f = 1$ MHz	
Timing Parameters							
Clock Frequency	f_{CLK}	—	—	3.6 1.35	MHz MHz	$V_{DD} = 5V$ (Note 3) $V_{DD} = 2.7V$ (Note 3)	
Clock High Time	t_{HI}	125	—	—	ns		
Clock Low Time	t_{LO}	125	—	—	ns		
\overline{CS} Fall To First Rising CLK Edge	t_{SUCS}	100	—	—	ns		
\overline{CS} Fall To Falling CLK Edge	t_{CSD}	—	—	0	ns		
Data Input Setup Time	t_{SU}	—	—	50	ns		
Data Input Hold Time	t_{HD}	—	—	50	ns		
CLK Fall To Output Data Valid	t_{DO}	—	—	125 200	ns ns	$V_{DD} = 5V$, See Figure 1-2 $V_{DD} = 2.7V$, See Figure 1-2	
CLK Fall To Output Enable	t_{EN}	—	—	125 200	ns ns	$V_{DD} = 5V$, See Figure 1-2 $V_{DD} = 2.7V$, See Figure 1-2	
\overline{CS} Rise To Output Disable	t_{DIS}	—	—	100	ns	See Test Circuits, Figure 1-2	
\overline{CS} Disable Time	t_{CSH}	270	—	—	ns		
D_{OUT} Rise Time	t_R	—	—	100	ns	See Test Circuits, Figure 1-2 (Note 1)	
D_{OUT} Fall Time	t_F	—	—	100	ns	See Test Circuits, Figure 1-2 (Note 1)	

Note 1: This parameter is established by characterization and not 100% tested.

2: See graphs that relate linearity performance to V_{REF} levels.

3: Because the sample cap will eventually lose charge, effective clock rates below 10 kHz can affect linearity performance, especially at elevated temperatures. See Section 6.2, "Maintaining Minimum Clock Speed", for more information.

MCP3004/3008

ELECTRICAL SPECIFICATIONS (CONTINUED)

Electrical Characteristics: Unless otherwise noted, all parameters apply at $V_{DD} = 5V$, $V_{REF} = 5V$, $T_{AMB} = -40^{\circ}C$ to $+85^{\circ}C$, $f_{SAMPLE} = 200$ kps and $f_{CLK} = 18 * f_{SAMPLE}$. Unless otherwise noted, typical values apply for $V_{DD} = 5V$, $T_{AMB} = 25^{\circ}C$.

Parameter	Sym	Min	Typ	Max	Units	Conditions
Power Requirements						
Operating Voltage	V_{DD}	2.7	—	5.5	V	
Operating Current	I_{DD}	—	425 225	550	μA	$V_{DD} = V_{REF} = 5V$, D_{OUT} unloaded $V_{DD} = V_{REF} = 2.7V$, D_{OUT} unloaded
Standby Current	I_{DDS}	—	0.005	2	μA	$\overline{CS} = V_{DD} = 5.0V$
Temperature Ranges						
Specified Temperature Range	T_A	-40	—	+85	$^{\circ}C$	
Operating Temperature Range	T_A	-40	—	+85	$^{\circ}C$	
Storage Temperature Range	T_A	-65	—	+150	$^{\circ}C$	
Thermal Package Resistance						
Thermal Resistance, 14L-PDIP	θ_{JA}	—	70	—	$^{\circ}C/W$	
Thermal Resistance, 14L-SOIC	θ_{JA}	—	108	—	$^{\circ}C/W$	
Thermal Resistance, 14L-TSSOP	θ_{JA}	—	100	—	$^{\circ}C/W$	
Thermal Resistance, 16L-PDIP	θ_{JA}	—	70	—	$^{\circ}C/W$	
Thermal Resistance, 16L-SOIC	θ_{JA}	—	90	—	$^{\circ}C/W$	

- Note 1:** This parameter is established by characterization and not 100% tested.
Note 2: See graphs that relate linearity performance to V_{REF} levels.
Note 3: Because the sample cap will eventually lose charge, effective clock rates below 10 kHz can affect linearity performance, especially at elevated temperatures. See Section 6.2, "Maintaining Minimum Clock Speed", for more information.

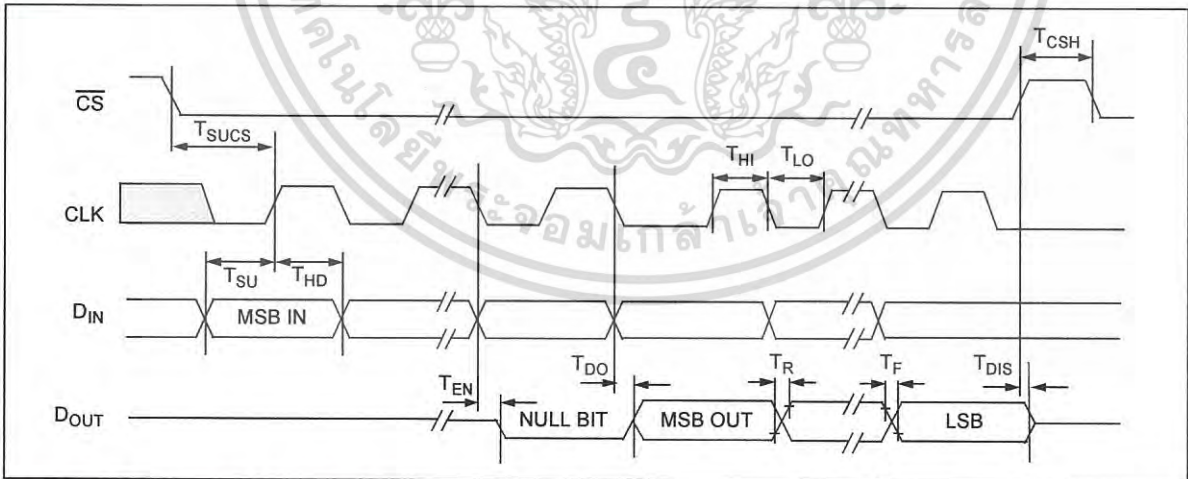


FIGURE 1-1: Serial Interface Timing.

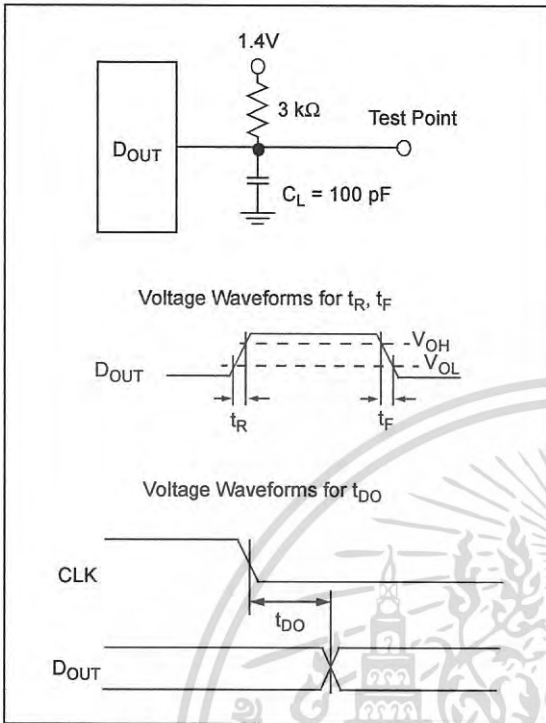


FIGURE 1-2: Load Circuit for t_R , t_F , t_{DO} .

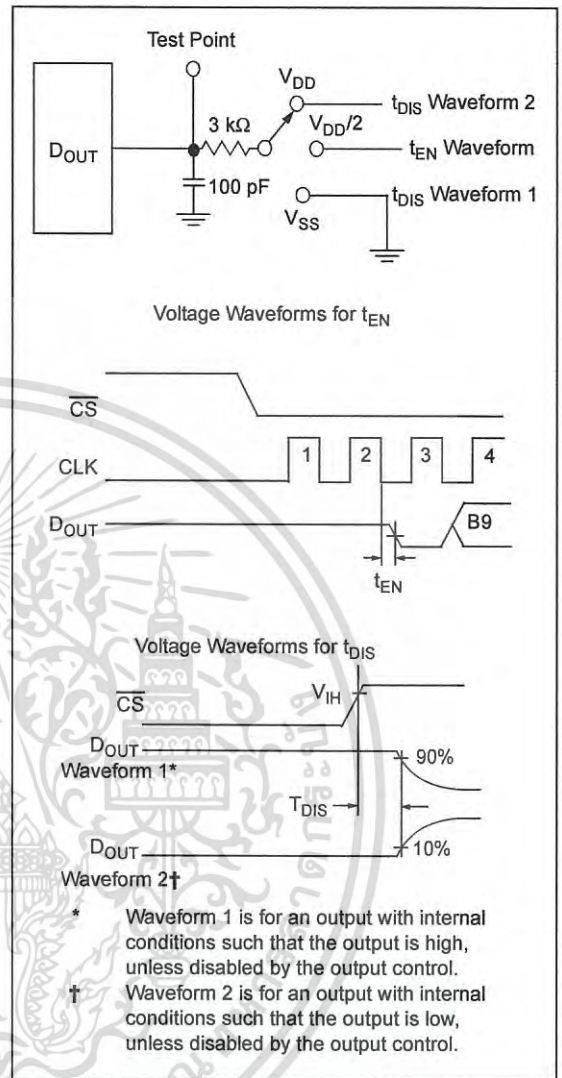


FIGURE 1-3: Load circuit for t_{DIS} and t_{EN} .

Fully Integrated, Hall Effect-Based Linear Current Sensor IC with 2.1 kVRMS Isolation and a Low-Resistance Current Conductor

Features and Benefits

- Low-noise analog signal path
- Device bandwidth is set via the new FILTER pin
- 5 μ s output rise time in response to step input current
- 80 kHz bandwidth
- Total output error 1.5% at $T_A = 25^\circ\text{C}$
- Small footprint, low-profile SOIC8 package
- 1.2 m Ω internal conductor resistance
- 2.1 kVRMS minimum isolation voltage from pins 1-4 to pins 5-8
- 5.0 V, single supply operation
- 66 to 185 mV/A output sensitivity
- Output voltage proportional to AC or DC currents
- Factory-trimmed for accuracy
- Extremely stable output offset voltage
- Nearly zero magnetic hysteresis
- Ratiometric output from supply voltage

Description

The Allegro™ ACS712 provides economical and precise solutions for AC or DC current sensing in industrial, commercial, and communications systems. The device package allows for easy implementation by the customer. Typical applications include motor control, load detection and management, switch-mode power supplies, and overcurrent fault protection. The device is not intended for automotive applications.

The device consists of a precise, low-offset, linear Hall circuit with a copper conduction path located near the surface of the die. Applied current flowing through this copper conduction path generates a magnetic field which the Hall IC converts into a proportional voltage. Device accuracy is optimized through the close proximity of the magnetic signal to the Hall transducer. A precise, proportional voltage is provided by the low-offset, chopper-stabilized BiCMOS Hall IC, which is programmed for accuracy after packaging.

The output of the device has a positive slope ($>V_{IOUT(Q)}$) when an increasing current flows through the primary copper conduction path (from pins 1 and 2, to pins 3 and 4), which is the path used for current sampling. The internal resistance of this conductive path is 1.2 m Ω typical, providing low power loss. The thickness of the copper conductor allows survival of

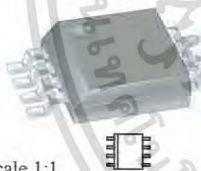
Continued on the next page...



TÜV America
Certificate Number
U8V 06 05.54214 010

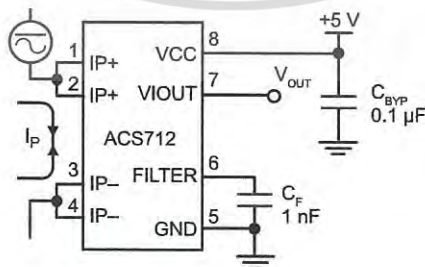


Package: 8 Lead SOIC (suffix LC)



Approximate Scale 1:1

Typical Application



Application 1. The ACS712 outputs an analog signal, V_{OUT} , that varies linearly with the uni- or bi-directional AC or DC primary sampled current, I_P , within the range specified. C_F is recommended for noise management, with values that depend on the application.

ACS712

Fully Integrated, Hall Effect-Based Linear Current Sensor IC with 2.1 kVRMS Isolation and a Low-Resistance Current Conductor

Description (continued)

the device at up to 5× overcurrent conditions. The terminals of the conductive path are electrically isolated from the signal leads (pins 5 through 8). This allows the ACS712 to be used in applications requiring electrical isolation without the use of opto-isolators or other costly isolation techniques.

The ACS712 is provided in a small, surface mount SOIC8 package. The leadframe is plated with 100% matte tin, which is compatible with standard lead (Pb) free printed circuit board assembly processes. Internally, the device is Pb-free, except for flip-chip high-temperature Pb-based solder balls, currently exempt from RoHS. The device is fully calibrated prior to shipment from the factory.

Selection Guide

Part Number	Packing*	T _A (°C)	Optimized Range, I _P (A)	Sensitivity, Sens (Typ) (mV/A)
ACS712ELCTR-05B-T	Tape and reel, 3000 pieces/reel	-40 to 85	±5	185
ACS712ELCTR-20A-T	Tape and reel, 3000 pieces/reel	-40 to 85	±20	100
ACS712ELCTR-30A-T	Tape and reel, 3000 pieces/reel	-40 to 85	±30	66

*Contact Allegro for additional packing options.

Absolute Maximum Ratings

Characteristic	Symbol	Notes	Rating	Units
Supply Voltage	V _{CC}		8	V
Reverse Supply Voltage	V _{RCC}		-0.1	V
Output Voltage	V _{IOUT}		8	V
Reverse Output Voltage	V _{RIOUT}		-0.1	V
Output Current Source	I _{IOUT(SOURCE)}		3	mA
Output Current Sink	I _{IOUT(SINK)}		10	mA
Overcurrent Transient Tolerance	I _P	1 pulse, 100 ms	100	A
Nominal Operating Ambient Temperature	T _A	Range E	-40 to 85	°C
Maximum Junction Temperature	T _{J(max)}		165	°C
Storage Temperature	T _{stg}		-65 to 170	°C

Isolation Characteristics

Characteristic	Symbol	Notes	Rating	Unit
Dielectric Strength Test Voltage*	V _{ISO}	Agency type-tested for 60 seconds per UL standard 60950-1, 1st Edition	2100	VAC
Working Voltage for Basic Isolation	V _{WFSI}	For basic (single) isolation per UL standard 60950-1, 1st Edition	354	VDC or V _{pk}
Working Voltage for Reinforced Isolation	V _{WFRI}	For reinforced (double) isolation per UL standard 60950-1, 1st Edition	184	VDC or V _{pk}

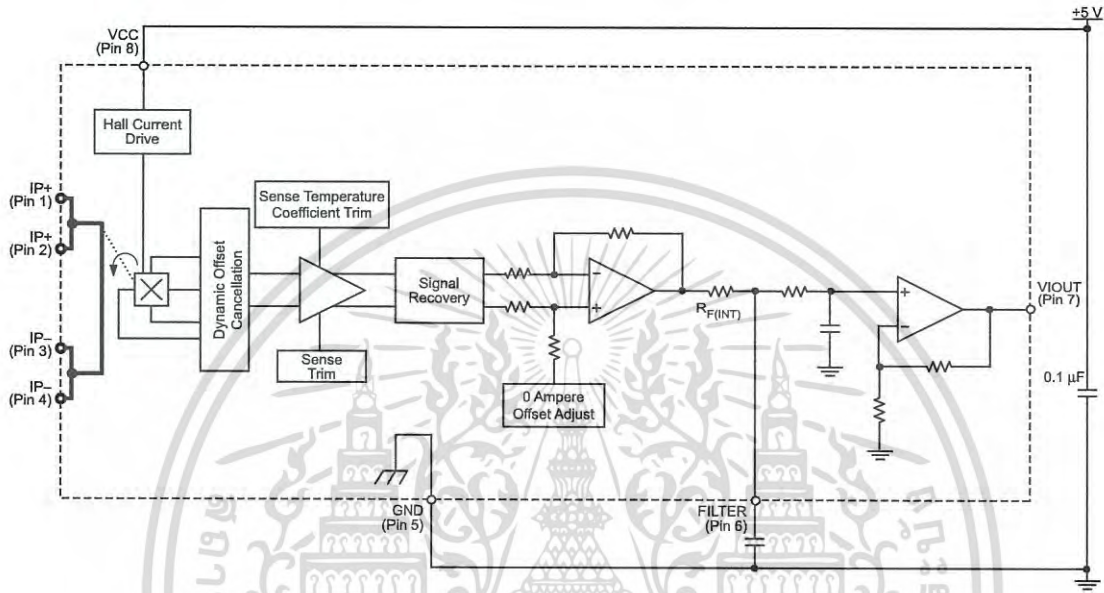
* Allegro does not conduct 60-second testing. It is done only during the UL certification process.

Parameter	Specification
Fire and Electric Shock	CAN/CSA-C22.2 No. 60950-1-03 UL 60950-1:2003 EN 60950-1:2001

ACS712

Fully Integrated, Hall Effect-Based Linear Current Sensor IC
with 2.1 kVRMS Isolation and a Low-Resistance Current Conductor

Functional Block Diagram



Pin-out Diagram



Terminal List Table

Number	Name	Description
1 and 2	IP+	Terminals for current being sampled; fused internally
3 and 4	IP-	Terminals for current being sampled; fused internally
5	GND	Signal ground terminal
6	FILTER	Terminal for external capacitor that sets bandwidth
7	VIOOUT	Analog output signal
8	VCC	Device power supply terminal

ACS712

Fully Integrated, Hall Effect-Based Linear Current Sensor IC with 2.1 kVRMS Isolation and a Low-Resistance Current Conductor

COMMON OPERATING CHARACTERISTICS¹ over full range of T_A , $C_F = 1$ nF, and $V_{CC} = 5$ V, unless otherwise specified

Characteristic	Symbol	Test Conditions	Min.	Typ.	Max.	Units
ELECTRICAL CHARACTERISTICS						
Supply Voltage	V_{CC}		4.5	5.0	5.5	V
Supply Current	I_{CC}	$V_{CC} = 5.0$ V, output open	–	10	13	mA
Output Capacitance Load	C_{LOAD}	V _{IOUT} to GND	–	–	10	nF
Output Resistive Load	R_{LOAD}	V _{IOUT} to GND	4.7	–	–	k Ω
Primary Conductor Resistance	$R_{PRIMARY}$	$T_A = 25^\circ\text{C}$	–	1.2	–	m Ω
Rise Time	t_r	$I_P = I_P(\text{max})$, $T_A = 25^\circ\text{C}$, $C_{OUT} = \text{open}$	–	3.5	–	μs
Frequency Bandwidth	f	–3 dB, $T_A = 25^\circ\text{C}$; I_P is 10 A peak-to-peak	–	80	–	kHz
Nonlinearity	E_{LIN}	Over full range of I_P	–	1.5	–	%
Symmetry	E_{SYM}	Over full range of I_P	98	100	102	%
Zero Current Output Voltage	$V_{IOUT(Q)}$	Bidirectional; $I_P = 0$ A, $T_A = 25^\circ\text{C}$	–	$V_{CC} \times 0.5$	–	V
Power-On Time	t_{PO}	Output reaches 90% of steady-state level, $T_J = 25^\circ\text{C}$, 20 A present on leadframe	–	35	–	μs
Magnetic Coupling ²			–	12	–	G/A
Internal Filter Resistance ³	$R_{F(INT)}$			1.7		k Ω

¹Device may be operated at higher primary current levels, I_P , and ambient, T_A , and internal leadframe temperatures, T_A , provided that the Maximum Junction Temperature, $T_J(\text{max})$, is not exceeded.

²1G = 0.1 mT.

³ $R_{F(INT)}$ forms an RC circuit via the FILTER pin.

COMMON THERMAL CHARACTERISTICS¹

Characteristic	Symbol	Test Conditions	Min.	Typ.	Max.	Units
Operating Internal Leadframe Temperature	T_A	E range	–40	–	85	$^\circ\text{C}$
Junction-to-Lead Thermal Resistance ²	$R_{\theta JL}$	Mounted on the Allegro ASEK 712 evaluation board			5	$^\circ\text{C/W}$
Junction-to-Ambient Thermal Resistance	$R_{\theta JA}$	Mounted on the Allegro 85-0322 evaluation board, includes the power consumed by the board			23	$^\circ\text{C/W}$

¹Additional thermal information is available on the Allegro website.

²The Allegro evaluation board has 1500 mm² of 2 oz. copper on each side, connected to pins 1 and 2, and to pins 3 and 4, with thermal vias connecting the layers. Performance values include the power consumed by the PCB. Further details on the board are available from the Frequently Asked Questions document on our website. Further information about board design and thermal performance also can be found in the Applications Information section of this datasheet.

ACS712

Fully Integrated, Hall Effect-Based Linear Current Sensor IC with 2.1 kVRMS Isolation and a Low-Resistance Current Conductor

x05B PERFORMANCE CHARACTERISTICS¹ $T_A = -40^\circ\text{C}$ to 85°C , $C_F = 1\text{ nF}$, and $V_{CC} = 5\text{ V}$, unless otherwise specified

Characteristic	Symbol	Test Conditions	Min.	Typ.	Max.	Units
Optimized Accuracy Range	I_P		-5	-	5	A
Sensitivity	Sens	Over full range of I_P , $T_A = 25^\circ\text{C}$	180	185	190	mV/A
Noise	$V_{\text{NOISE(PP)}}$	Peak-to-peak, $T_A = 25^\circ\text{C}$, 185 mV/A programmed Sensitivity, $C_F = 47\text{ nF}$, $C_{\text{OUT}} = \text{open}$, 2 kHz bandwidth	-	21	-	mV
Zero Current Output Slope	$\Delta V_{\text{OUT(Q)}}$	$T_A = -40^\circ\text{C}$ to 25°C	-	-0.26	-	mV/°C
		$T_A = 25^\circ\text{C}$ to 150°C	-	-0.08	-	mV/°C
Sensitivity Slope	ΔSens	$T_A = -40^\circ\text{C}$ to 25°C	-	0.054	-	mV/A/°C
		$T_A = 25^\circ\text{C}$ to 150°C	-	-0.008	-	mV/A/°C
Total Output Error ²	E_{TOT}	$I_P = \pm 5\text{ A}$, $T_A = 25^\circ\text{C}$	-	± 1.5	-	%

¹Device may be operated at higher primary current levels, I_P , and ambient temperatures, T_A , provided that the Maximum Junction Temperature, $T_{J(\text{max})}$, is not exceeded.

²Percentage of I_P , with $I_P = 5\text{ A}$. Output filtered.

x20A PERFORMANCE CHARACTERISTICS¹ $T_A = -40^\circ\text{C}$ to 85°C , $C_F = 1\text{ nF}$, and $V_{CC} = 5\text{ V}$, unless otherwise specified

Characteristic	Symbol	Test Conditions	Min.	Typ.	Max.	Units
Optimized Accuracy Range	I_P		-20	-	20	A
Sensitivity	Sens	Over full range of I_P , $T_A = 25^\circ\text{C}$	96	100	104	mV/A
Noise	$V_{\text{NOISE(PP)}}$	Peak-to-peak, $T_A = 25^\circ\text{C}$, 100 mV/A programmed Sensitivity, $C_F = 47\text{ nF}$, $C_{\text{OUT}} = \text{open}$, 2 kHz bandwidth	-	11	-	mV
Zero Current Output Slope	$\Delta V_{\text{OUT(Q)}}$	$T_A = -40^\circ\text{C}$ to 25°C	-	-0.34	-	mV/°C
		$T_A = 25^\circ\text{C}$ to 150°C	-	-0.07	-	mV/°C
Sensitivity Slope	ΔSens	$T_A = -40^\circ\text{C}$ to 25°C	-	0.017	-	mV/A/°C
		$T_A = 25^\circ\text{C}$ to 150°C	-	-0.004	-	mV/A/°C
Total Output Error ²	E_{TOT}	$I_P = \pm 20\text{ A}$, $T_A = 25^\circ\text{C}$	-	± 1.5	-	%

¹Device may be operated at higher primary current levels, I_P , and ambient temperatures, T_A , provided that the Maximum Junction Temperature, $T_{J(\text{max})}$, is not exceeded.

²Percentage of I_P , with $I_P = 20\text{ A}$. Output filtered.

x30A PERFORMANCE CHARACTERISTICS¹ $T_A = -40^\circ\text{C}$ to 85°C , $C_F = 1\text{ nF}$, and $V_{CC} = 5\text{ V}$, unless otherwise specified

Characteristic	Symbol	Test Conditions	Min.	Typ.	Max.	Units
Optimized Accuracy Range	I_P		-30	-	30	A
Sensitivity	Sens	Over full range of I_P , $T_A = 25^\circ\text{C}$	63	66	69	mV/A
Noise	$V_{\text{NOISE(PP)}}$	Peak-to-peak, $T_A = 25^\circ\text{C}$, 66 mV/A programmed Sensitivity, $C_F = 47\text{ nF}$, $C_{\text{OUT}} = \text{open}$, 2 kHz bandwidth	-	7	-	mV
Zero Current Output Slope	$\Delta V_{\text{OUT(Q)}}$	$T_A = -40^\circ\text{C}$ to 25°C	-	-0.35	-	mV/°C
		$T_A = 25^\circ\text{C}$ to 150°C	-	-0.08	-	mV/°C
Sensitivity Slope	ΔSens	$T_A = -40^\circ\text{C}$ to 25°C	-	0.007	-	mV/A/°C
		$T_A = 25^\circ\text{C}$ to 150°C	-	-0.002	-	mV/A/°C
Total Output Error ²	E_{TOT}	$I_P = \pm 30\text{ A}$, $T_A = 25^\circ\text{C}$	-	± 1.5	-	%

¹Device may be operated at higher primary current levels, I_P , and ambient temperatures, T_A , provided that the Maximum Junction Temperature, $T_{J(\text{max})}$, is not exceeded.

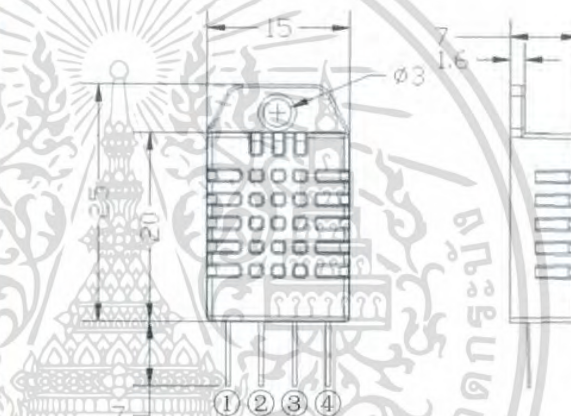
²Percentage of I_P , with $I_P = 30\text{ A}$. Output filtered.

1、 Product Overview

AM2302 capacitive humidity sensing digital temperature and humidity module is one that contains the compound has been calibrated digital signal output of the temperature and humidity sensors. Application of a dedicated digital modules collection technology and the temperature and humidity sensing technology, to ensure that the product has high reliability and excellent long-term stability. The sensor includes a capacitive sensor wet components and a high-precision temperature measurement devices, and connected with a high-performance 8-bit microcontroller. The product has excellent quality, fast response, strong anti-jamming capability, and high cost. Each sensor is extremely accurate humidity calibration chamber calibration. The form of procedures, the calibration coefficients stored in the microcontroller, the sensor within the processing of the heartbeat to call these calibration coefficients. Standard single-bus interface, system integration quick and easy. Small size, low power consumption, signal transmission distance up to 20 meters, making it the best choice of all kinds of applications and even the most demanding applications. Products for the 3-lead (single-bus interface) connection convenience. Special packages according to user needs.



Physical map



Dimensions (unit: mm)

2、 Applications

HVAC, dehumidifier, testing and inspection equipment, consumer goods, automotive, automatic control, data loggers, home appliances, humidity regulator, medical, weather stations, and other humidity measurement and control and so on.

3、 Features

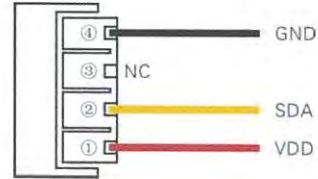
Ultra-low power, the transmission distance, fully automated calibration, the use of capacitive humidity sensor, completely interchangeable, standard digital single-bus output, excellent long-term stability, high accuracy temperature measurement devices.

4、 The definition of single-bus interface

4.1 AM2302 Pin assignments

Table 1: AM2302 Pin assignments

Pin	Name	Description
①	VDD	Power (3.3V-5.5V)
②	SDA	Serial data, bidirectional port
③	NC	Empty
④	GND	Ground



PIC1: AM2302 Pin Assignment

4.2 Power supply pins (VDD GND)

AM2302 supply voltage range 3.3V – 5.5V, recommended supply voltage is 5V.

4.3 Serial data (SDA)

SDA pin is tri structure for reading, writing sensor data. Specific communication timing, see the detailed description of the communication protocol.

5、 Sensor performance

5.1 Relative humidity

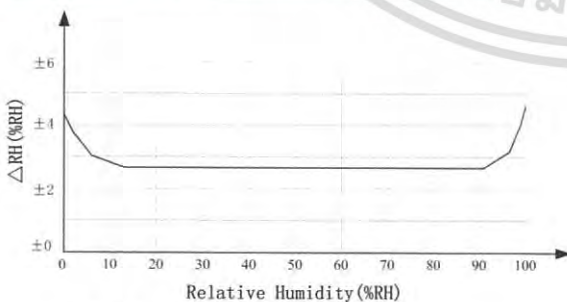
Table 2: AM2302 Relative humidity performance table

Parameter	Condition	min	typ	max	Unit
Resolution			0.1		%RH
Range		0		99.9	%RH
Accuracy ^[1]	25°C		± 2		%RH
Repeatability			± 0.3		%RH
Exchange	Completely interchangeable				
Response ^[2]	1/e(63%)		<5		S
Sluggish			<0.3		%RH
Drift ^[3]	Typical		<0.5		%RH/yr

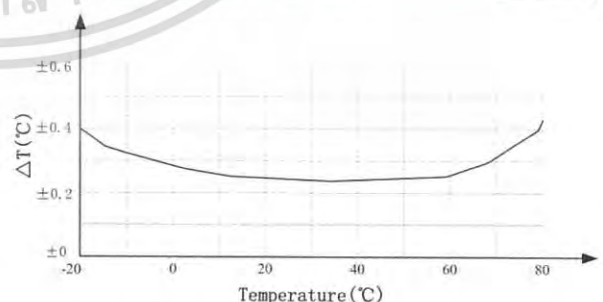
5.2 Temperature

Table 3: AM2302 Relative temperature performance

Parameter	Condition	min	typ	max	Unit
Resolution			0.1		°C
n			16		bit
Accuracy			± 0.5	± 1	°C
Range		-40		80	°C
Repeat			± 0.2		°C
Exchange	Completely interchangeable				
Response	1/e(63%)		<10		S
Drift			± 0.3		°C/yr



Pic2: At25°C The error of relative humidity



Pic3: The maximum temperature error

6、Electrical Characteristics

Electrical characteristics, such as energy consumption, high, low, input, output voltage, depending on the power supply. Table 4 details the electrical characteristics of the AM2302, if not identified, said supply voltage of 5V. To get the best results with the sensor, please design strictly in accordance with the conditions of design in Table 4.

Table 4: AM2302 DC Characteristics

Parameter	Condition	min	typ	max	Unit
Voltage		3.3	5	5.5	V
Power consumption ^[4]	Dormancy	10	15		μA
	Measuring		500		μA
	Average		300		μA
Low level output voltage	I _{OL} ^[5]	0		300	mV
High output voltage	R _p <25 kΩ	90%		100%	VDD
Low input voltage	Decline	0		30%	VDD
Input High Voltage	Rise	70%		100%	VDD
R _{pu} ^[6]	VDD = 5V VIN = VSS	30	45	60	kΩ
Output current	turn on		8		mA
	turn off	10	20		μA
Sampling period		2			S

[1] the accuracy of the factory inspection, the sensor 25°C and 5V, the accuracy specification of test conditions, it does not include hysteresis and nonlinearity, and is only suitable for non-condensing environment.

[2] to achieve an order of 63% of the time required under the conditions of 25°C and 1m / s airflow.

[3] in the volatile organic compounds, the values may be higher. See the manual application to store information.

[4] this value at VDD = 5.0V when the temperature is 25°C, 2S / time, under the conditions of the average.

[5] low output current.

[6] that the pull-up resistor.

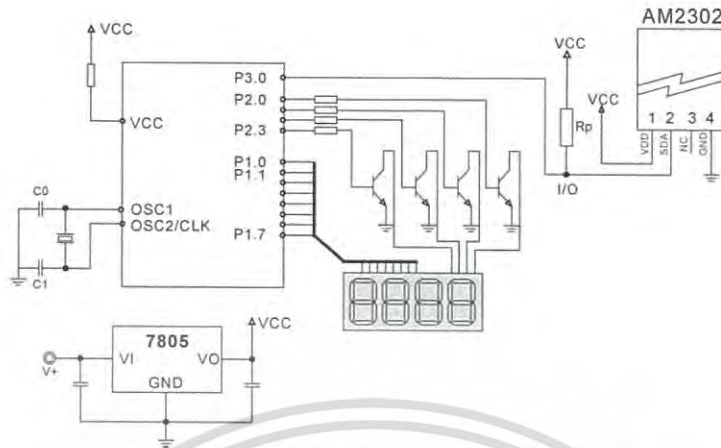
7、Single-bus communication (ONE-WIRE)

7.1 Typical circuits for single bus

Microprocessor and AM2302 connection typical application circuit is shown in Figure 4. Single bus communication mode, pull the SDA microprocessor I / O port is connected.

Special instructions of the single-bus communication:

1. Typical application circuit recommended in the short cable length of 30 meters on the 5.1K pull-up resistor pullup resistor according to the actual situation of lower than 30 m.
2. With 3.3V supply voltage, cable length shall not be greater than 100cm. Otherwise, the line voltage drop will lead to the sensor power supply, resulting in measurement error.
3. Read the sensor minimum time interval for the 2S; read interval is less than 2S, may cause the temperature and humidity are not allowed or communication is unsuccessful, etc..
4. Temperature and humidity values are each read out the results of the last measurement For real-time data that need continuous read twice, we recommend repeatedly to read sensors, and each read sensor interval is greater than 2 seconds to obtain accuratethe data.



Pic4: AM2302 Typical circuits for single bus

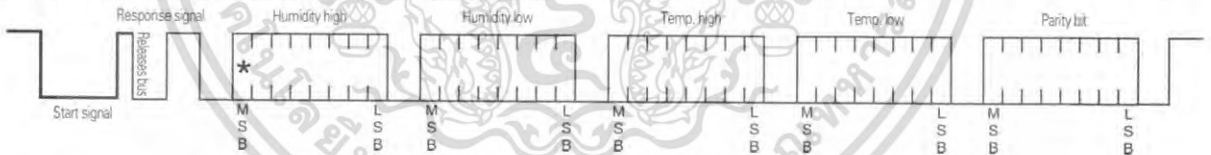
7.2、 Single-bus communication protocol

◎ Single bus Description

AM2302 device uses a simplified single-bus communication. Single bus that only one data line, data exchange system, controlled by the data line to complete. Equipment (microprocessor) through an open-drain or tri-state port connected to the data line to allow the device does not send data to release the bus, while other devices use the bus; single bus usually require an external about 5.1kΩ pull-up resistor, so when the bus is idle, its status is high. Because they are the master-slave structure, only the host calls the sensor, the sensor will answer, so the hosts to access the sensor must strictly follow the sequence of single bus, if there is a sequence of confusion, the sensor will not respond to the host.

◎ Single bus to send data definition

SDA For communication and synchronization between the microprocessor and the AM2302, single-bus data format, a transmission of 40 data, the high first-out. Specific communication timing shown in Figure 5, the communication signal format is depicted in Table 5.



Pic5: AM2302 Single-bus communication protocol

General Description

The DS3231 is a low-cost, extremely accurate I²C real-time clock (RTC) with an integrated temperature-compensated crystal oscillator (TCXO) and crystal. The device incorporates a battery input, and maintains accurate timekeeping when main power to the device is interrupted. The integration of the crystal resonator enhances the long-term accuracy of the device as well as reduces the piece-part count in a manufacturing line. The DS3231 is available in commercial and industrial temperature ranges, and is offered in a 16-pin, 300-mil SO package.

The RTC maintains seconds, minutes, hours, day, date, month, and year information. The date at the end of the month is automatically adjusted for months with fewer than 31 days, including corrections for leap year. The clock operates in either the 24-hour or 12-hour format with an AM/PM indicator. Two programmable time-of-day alarms and a programmable square-wave output are provided. Address and data are transferred serially through an I²C bidirectional bus.

A precision temperature-compensated voltage reference and comparator circuit monitors the status of V_{CC} to detect power failures, to provide a reset output, and to automatically switch to the backup supply when necessary. Additionally, the RST pin is monitored as a pushbutton input for generating a µP reset.

Benefits and Features

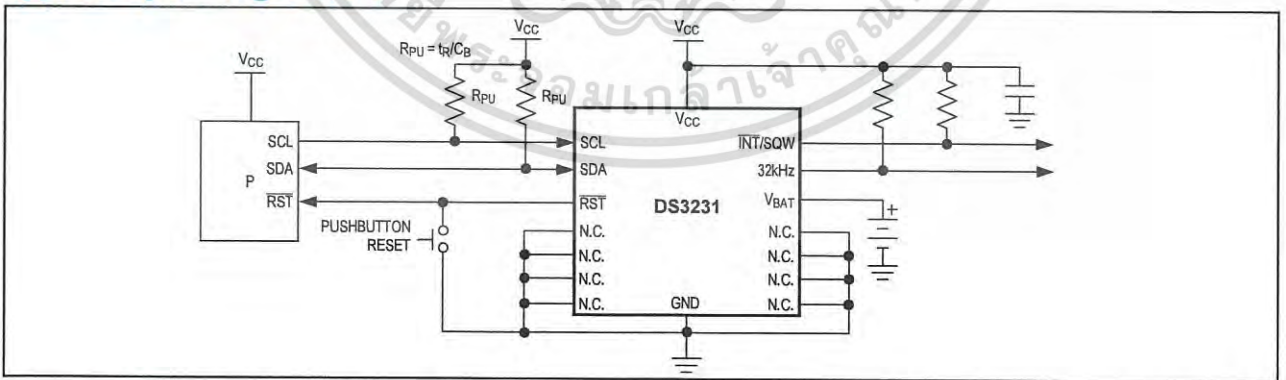
- Highly Accurate RTC Completely Manages All Timekeeping Functions
 - Real-Time Clock Counts Seconds, Minutes, Hours, Date of the Month, Month, Day of the Week, and Year, with Leap-Year Compensation Valid Up to 2100
 - Accuracy ±2ppm from 0°C to +40°C
 - Accuracy ±3.5ppm from -40°C to +85°C
 - Digital Temp Sensor Output: ±3°C Accuracy
 - Register for Aging Trim
 - RST Output/Pushbutton Reset Debounce Input
 - Two Time-of-Day Alarms
 - Programmable Square-Wave Output Signal
- Simple Serial Interface Connects to Most Microcontrollers
 - Fast (400kHz) I²C Interface
- Battery-Backup Input for Continuous Timekeeping
 - Low Power Operation Extends Battery-Backup Run Time
 - 3.3V Operation
- Operating Temperature Ranges: Commercial (0°C to +70°C) and Industrial (-40°C to +85°C)
- Underwriters Laboratories® (UL) Recognized

Applications

- Servers
- Telematics
- Utility Power Meters
- GPS

Ordering Information and Pin Configuration appear at end of data sheet.

Typical Operating Circuit



Underwriters Laboratories is a registered certification mark of Underwriters Laboratories Inc.

Absolute Maximum Ratings

Voltage Range on Any Pin Relative to Ground-0.3V to +6.0V
 Junction-to-Ambient Thermal Resistance (θ_{JA}) (Note 1) 73°C/W
 Junction-to-Case Thermal Resistance (θ_{JC}) (Note 1)23°C/W
 Operating Temperature Range
 DS3231S0°C to +70°C
 DS3231SN..... -40°C to +85°C

Junction Temperature+125°C
 Storage Temperature Range -40°C to +85°C
 Lead Temperature (soldering, 10s)+260°C
 Soldering Temperature (reflow, 2 times max)+260°C
 (see the *Handling, PCB Layout, and Assembly* section)

Note 1: Package thermal resistances were obtained using the method described in JEDEC specification JESD51-7, using a four-layer board. For detailed information on package thermal considerations, refer to www.maximintegrated.com/thermal-tutorial.

Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of the specifications is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Recommended Operating Conditions

($T_A = T_{MIN}$ to T_{MAX} , unless otherwise noted.) (Notes 2, 3)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
Supply Voltage	V_{CC}		2.3	3.3	5.5	V
	V_{BAT}		2.3	3.0	5.5	V
Logic 1 Input SDA, SCL	V_{IH}		0.7 x V_{CC}		$V_{CC} + 0.3$	V
Logic 0 Input SDA, SCL	V_{IL}		-0.3		0.3 x V_{CC}	V

Electrical Characteristics

($V_{CC} = 2.3V$ to $5.5V$, V_{CC} = Active Supply (see Table 1), $T_A = T_{MIN}$ to T_{MAX} , unless otherwise noted.) (Typical values are at $V_{CC} = 3.3V$, $V_{BAT} = 3.0V$, and $T_A = +25^\circ C$, unless otherwise noted.) (Notes 2, 3)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
Active Supply Current	I_{CCA}	(Notes 4, 5)	$V_{CC} = 3.63V$		200	μA
			$V_{CC} = 5.5V$		300	
Standby Supply Current	I_{CCS}	I ² C bus inactive, 32kHz output on, SQW output off (Note 5)	$V_{CC} = 3.63V$		110	μA
			$V_{CC} = 5.5V$		170	
Temperature Conversion Current	$I_{CCSCONV}$	I ² C bus inactive, 32kHz output on, SQW output off	$V_{CC} = 3.63V$		575	μA
			$V_{CC} = 5.5V$		650	
Power-Fail Voltage	V_{PF}		2.45	2.575	2.70	V
Logic 0 Output, 32kHz, \overline{INT}/SQW , SDA	V_{OL}	$I_{OL} = 3mA$			0.4	V
Logic 0 Output, \overline{RST}	V_{OL}	$I_{OL} = 1mA$			0.4	V
Output Leakage Current 32kHz, \overline{INT}/SQW , SDA	I_{LO}	Output high impedance	-1	0	+1	μA
Input Leakage SCL	I_{LI}		-1		+1	μA
\overline{RST} Pin I/O Leakage	I_{OL}	\overline{RST} high impedance (Note 6)	-200		+10	μA
V_{BAT} Leakage Current (V_{CC} Active)	I_{BATLKG}			25	100	nA

Electrical Characteristics (continued)

(V_{CC} = 2.3V to 5.5V, V_{CC} = Active Supply (see Table 1), T_A = T_{MIN} to T_{MAX}, unless otherwise noted.) (Typical values are at V_{CC} = 3.3V, V_{BAT} = 3.0V, and T_A = +25°C, unless otherwise noted.) (Notes 2, 3)

PARAMETER	SYMBOL	CONDITIONS		MIN	TYP	MAX	UNITS
Output Frequency	f _{OUT}	V _{CC} = 3.3V or V _{BAT} = 3.3V			32.768		kHz
Frequency Stability vs. Temperature (Commercial)	Δf/f _{OUT}	V _{CC} = 3.3V or V _{BAT} = 3.3V, aging offset = 00h	0°C to +40°C			±2	ppm
			>40°C to +70°C			±3.5	
Frequency Stability vs. Temperature (Industrial)	Δf/f _{OUT}	V _{CC} = 3.3V or V _{BAT} = 3.3V, aging offset = 00h	-40°C to <0°C			±3.5	ppm
			0°C to +40°C			±2	
			>40°C to +85°C			±3.5	
Frequency Stability vs. Voltage	Δf/V				1		ppm/V
Trim Register Frequency Sensitivity per LSB	Δf/LSB	Specified at:	-40°C			0.7	ppm
			+25°C			0.1	
			+70°C			0.4	
			+85°C			0.8	
Temperature Accuracy	Temp	V _{CC} = 3.3V or V _{BAT} = 3.3V		-3		+3	°C
Crystal Aging	Δf/f _O	After reflow, not production tested	First year			±1.0	ppm
			0–10 years			±5.0	

Electrical Characteristics

(V_{CC} = 0V, V_{BAT} = 2.3V to 5.5V, T_A = T_{MIN} to T_{MAX}, unless otherwise noted.) (Note 2)

PARAMETER	SYMBOL	CONDITIONS		MIN	TYP	MAX	UNITS
Active Battery Current	I _{BATA}	E _{OSC} = 0, BBSQW = 0, SCL = 400kHz (Note 5)	V _{BAT} = 3.63V			70	μA
			V _{BAT} = 5.5V			150	
Timekeeping Battery Current	I _{BATT}	E _{OSC} = 0, BBSQW = 0, EN32kHz = 1, SCL = SDA = 0V or SCL = SDA = V _{BAT} (Note 5)	V _{BAT} = 3.63V		0.84	3.0	μA
			V _{BAT} = 5.5V		1.0	3.5	
Temperature Conversion Current	I _{BATTCC}	E _{OSC} = 0, BBSQW = 0, SCL = SDA = 0V or SCL = SDA = V _{BAT}	V _{BAT} = 3.63V			575	μA
			V _{BAT} = 5.5V			650	
Data-Retention Current	I _{BATTDR}	E _{OSC} = 1, SCL = SDA = 0V, +25°C				100	nA

AC Electrical Characteristics(V_{CC} = V_{CC(MIN)} to V_{CC(MAX)} or V_{BAT} = V_{BAT(MIN)} to V_{BAT(MAX)}, V_{BAT} > V_{CC}, T_A = T_{MIN} to T_{MAX}, unless otherwise noted.) (Note 2)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
SCL Clock Frequency	f _{SCL}	Fast mode	100		400	kHz
		Standard mode	0		100	
Bus Free Time Between STOP and START Conditions	t _{BUF}	Fast mode	1.3			μs
		Standard mode	4.7			
Hold Time (Repeated) START Condition (Note 7)	t _{HD:STA}	Fast mode	0.6			μs
		Standard mode	4.0			
Low Period of SCL Clock	t _{LOW}	Fast mode	1.3			μs
		Standard mode	4.7			
High Period of SCL Clock	t _{HIGH}	Fast mode	0.6			μs
		Standard mode	4.0			
Data Hold Time (Notes 8, 9)	t _{HD:DAT}	Fast mode	0		0.9	μs
		Standard mode	0		0.9	
Data Setup Time (Note 10)	t _{SU:DAT}	Fast mode	100			ns
		Standard mode	250			
START Setup Time	t _{SU:STA}	Fast mode	0.6			μs
		Standard mode	4.7			
Rise Time of Both SDA and SCL Signals (Note 11)	t _R	Fast mode	20 +		300	ns
		Standard mode	0.1C _B		1000	
Fall Time of Both SDA and SCL Signals (Note 11)	t _F	Fast mode	20 +		300	ns
		Standard mode	0.1C _B		300	
Setup Time for STOP Condition	t _{SU:STO}	Fast mode	0.6			μs
		Standard mode	4.7			
Capacitive Load for Each Bus Line	C _B	(Note 11)			400	pF
Capacitance for SDA, SCL	C _{I/O}			10		pF
Pulse Width of Spikes That Must Be Suppressed by the Input Filter	t _{SP}			30		ns
Pushbutton Debounce	PB _{DB}			250		ms
Reset Active Time	t _{RST}			250		ms
Oscillator Stop Flag (OSF) Delay	t _{OSF}	(Note 12)		100		ms
Temperature Conversion Time	t _{CONV}			125	200	ms

Power-Switch Characteristics(T_A = T_{MIN} to T_{MAX})

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
V _{CC} Fall Time; V _{PF(MAX)} to V _{PF(MIN)}	t _{VCCF}		300			μs
V _{CC} Rise Time; V _{PF(MIN)} to V _{PF(MAX)}	t _{VCCR}		0			μs
Recovery at Power-Up	t _{REC}	(Note 13)		250	300	ms