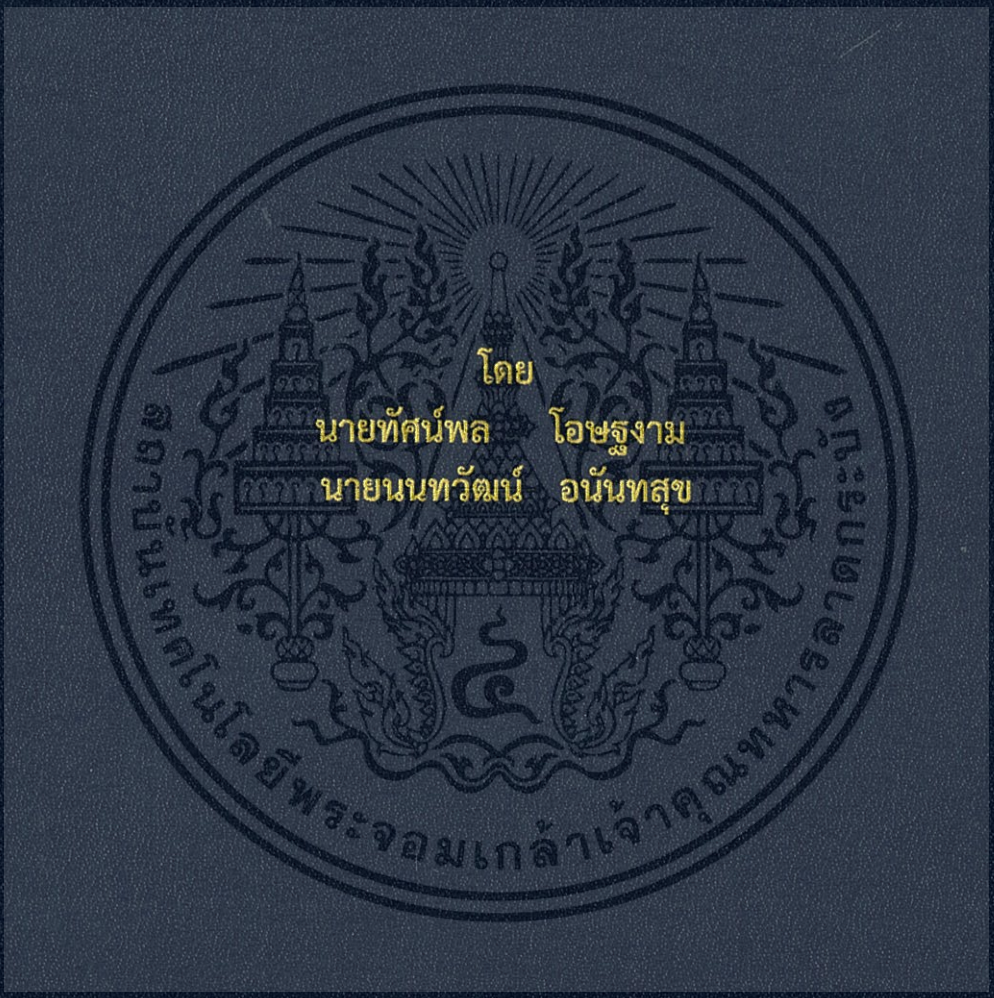
ระบบควบคุมการเฝ้าระวังในฟาร์มสัตว์
ANIMAL FARM SURVEILLANCE SYSTEM



โดย

นายทัศน์พล โอษฐงาม

นายนนทวัฒน์ อนันทสุข

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2560

ระบบควบคุมการเฝ้าระวังในฟาร์มสัตว์

Animal Farm Surveillance System

โดย

นายทัศน์พล โอษฐงาม 57010519

นายนนทวัฒน์ อนันทสุข 57010650

อาจารย์ที่ปรึกษา

ผศ.ดร. สมเกียรติ ฤกษ์วรัญญู

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2560

ผ่านการตรวจรูปเล่มแล้ว

(.....)

อาจารย์ที่ปรึกษา

21/5/61

KMITL วิศวกรรมโทรคมนาคม Telecommunications Engineering

ผ่านการตรวจชิ้นงานแล้ว

(.....)

กรรมการผู้ตรวจชิ้นงาน

21/5/61

KMITL วิศวกรรมโทรคมนาคม Telecommunications Engineering

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2560

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ระบบควบคุมและเฝ้าระวังในฟาร์มสัตว์
(ANIMAL FARM SURVEILLANCE SYSTEM)

ผู้จัดทำ

1. นายทัศน์พล โอษฐงาม 57010519
2. นายนนทวัฒน์ อนันทสุข 57010650

..... **อาจารย์ที่ปรึกษา**

(ผศ.ดร. สมเกียรติ ฤกษ์วรัญญู)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์นี้สำเร็จได้ด้วยดี เนื่องจากได้รับความกรุณาอย่างสูงจากผู้ช่วยศาสตร์ ดร.สมเกียรติ ฤกษ์วรัญญู และ ผู้ช่วยศาสตร์ ดร.มนตรี คำเงิน ในการให้คำแนะนำคำปรึกษา และตรวจดูความเรียบร้อยของโครงงานในการทำงาน ตลอดจนปรับปรุงแก้ไขข้อบกพร่องต่าง ๆ ด้วยความเอาใจใส่ และ ดูแลอย่างดียิ่ง คณะผู้จัดทำโครงงานนั้นได้ตระหนักถึงความตั้งใจจริงและความกรุณาที่ให้การช่วยเหลือตลอดระยะเวลาที่ทำโครงงานของอาจารย์จึงขอกราบขอบพระคุณเป็นอย่างสูงไว้ ณ ที่นี้

ขอขอบพระคุณอาจารย์ ภาควิชาวิศวกรรมโทรคมนาคม คณะวิศวกรรมศาสตร์สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบังที่ได้ประสิทธิ์ประสาทวิชาให้กับข้าพเจ้าขอขอบคุณเพื่อนๆพี่ๆน้องๆทุกคนที่คอยให้ความช่วยเหลือและห่วงใยโดยตลอดมาจนทำให้ปริญญานิพนธ์ฉบับนี้ลุล่วงมาได้

และอันดับสุดท้ายต้องกราบขอบพระคุณบุคคลที่สำคัญยิ่งที่ทำให้ข้าพเจ้ามีวันนี้ได้ก็คือ บิดา มารดา ที่เป็นผู้ให้กำเนิดและเลี้ยงดูข้าพเจ้าเป็นอย่างดี พร้อมทั้งให้โอกาสในการศึกษาอย่างเต็มที่ และเอาใจใส่ให้ความช่วยเหลือในทุกๆด้านเป็นอย่างดี

นายทัศน์พล โอษฐงาม
นายนนทวัฒน์ อนันทสุข
ผู้จัดทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบควบคุมการเฝ้าระวังในฟาร์มสัตว์
ANIMAL FARM SURVEILLANCE SYSTEM

โดย นายทัศน์พล โอษฐงาม 57010519
นายนนทวัฒน์ อนันทสุข 57010650

อาจารย์ที่ปรึกษา ผศ.ดร. สมเกียรติ ฤกษ์วรัญญู

บทคัดย่อ

ปริญญานิพนธ์นี้ได้ศึกษา การออกแบบ การสร้าง การทดสอบ และแสดงผลระบบควบคุมการเฝ้าระวังในฟาร์มสัตว์โดยได้ทำการศึกษาการทำฟาร์มสัตว์ในรูปของโมเดล โดยสัตว์ที่เลือกทำการศึกษาคือฟาร์มไก่เนื้อระบบปิด ได้ทำการศึกษาวิธีการเลี้ยง การดูแล ของฟาร์มไก่เนื้อระบบปิดเพื่อนำมาทำโมเดลจำลอง เพื่อให้สะดวกในการดูแลสัตว์ในฟาร์มให้อยู่ในสภาพแวดล้อมที่เหมาะสมโดยที่สามารถติดตามตามค่าความชื้นและอุณหภูมิในช่วงเวลาขณะนั้นได้ในทันที สามารถเปิดปิดอุปกรณ์ไฟฟ้าที่สำคัญภายในฟาร์มสัตว์ โดยหน้าแสดงผลและการควบคุมจะอยู่กับอุปกรณ์ SMART PHONE เพื่อเพิ่มความสะดวกในการควบคุมและดูแลภายในฟาร์มได้โดยที่ไม่จำเป็นต้องอยู่ในฟาร์มและได้ทำการเลือกอุปกรณ์ไมโครคอนโทรลเลอร์ที่ต้องใช้ในการทำโครงงานคร่าวๆ เช่น ARDUINO , DHT22 , RELAY , NODEMCU , SMART PHONE และศึกษาโปรแกรมที่ใช้ในการทำงาน และเพื่อป้องกันการเปลี่ยนแปลงอุณหภูมิที่มีผลกระทบกับสัตว์ และเฝ้าระวังภายในฟาร์มสัตว์โดยมีการแจ้งเตือน ALARM ไปยัง SMART PHONE เมื่อถึงค่าที่กำหนดเอาไว้

ABSTRACT

This thesis has studied the design, build, test and display control, surveillance, farm animals, farm animals, was studied as a model. The animal study was chosen chicken meat off. The study how to raise chickens to take care of the

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

farm system off to bring a model. In order to facilitate the care of farm animals in the environment that can be tracked by the humidity and temperature in that moment right away. You can turn off the power to the farm animals. The front display and control the device SMART PHONE to facilitate the control and supervision within the farm without the need to stay on the farm and select the device microcontroller to be used in the project brief example. ARDUINO, DHT22, RELAY, NODEMCU, SMART PHONE and education programs in the workplace. And to prevent temperature changes affect the animals. And surveillance on farm animal is alert to the SMART PHONE ALARM on the value assigned to it.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

		หน้า
กิตติกรรมประกาศ		I
บทคัดย่อ		II
สารบัญ		IV
สารบัญรูป		VI
บทที่ 1	**บทนำ**	**1**
	1.1 ความเป็นมาและความสำคัญของปัญหา	1
	1.2 วัตถุประสงค์	1
	1.3 ขอบเขตของโครงงาน	1
บทที่ 2	**ทฤษฎีและหลักการที่เกี่ยวข้อง**	**3**
	2.1 ความรู้เกี่ยวกับโรงเรือนระบบฟาร์มปิด	3
	2.2 คลื่นอัลตร้าโซนิค (ULTRA-SONIC WAVE)	7
	2.3 ARDUINO IDE	11
	2.4 ไมโครคอนโทรลเลอร์	12
	2.5 RELAY SWITCH	15
	2.6 โมดูลวัดอุณหภูมิและความชื้น	18
	2.7 RC SERVO MOTOR	21
	2.8 FIREBASE	24
	2.9 ANDROID STUDIO	26
	2.10 ภาษาจาวา (JAVA)	28
บทที่ 3	**การออกแบบและการจัดทำปริญญานิพนธ์**	**29**
	3.1 การออกแบบ	29
	3.2 เครื่องมือที่ใช้ในการทดลอง	48

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

	3.3 การจัดเก็บผลการทดลอง	49
บทที่ 4	**ผลการทดลอง**	**50**
	4.1 การวัดค่าอุณหภูมิและความชื้น	50
	4.2 การทดสอบการแสดงผลค่าอุณหภูมิบน SMART PHONE	54
	4.3 การทดสอบการเปิด-ปิดด้วยแอพพลิเคชั่น	55
	4.4 การทดสอบการวัดค่าระดับน้ำ	59
	4.5 การทดสอบการแจ้งเตือน	61
บทที่ 5	**สรุปผลและข้อเสนอแนะ**	**71**
	5.1 สรุปผล	71
	5.2 ข้อเสนอแนะ	71
บรรณานุกรม		**72**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่		หน้า
1.1	บล็อกไดอะแกรมของปริญญานิพนธ์	2
2.1	การทำงานของระบบ EVAP [1]	3
2.2	หลักการทำและวิธีการทำงานของ COOLING PAD [1]	4
2.3	คลื่นอัลตร้าโซนิค	7
2.4	เซนเซอร์อัลตร้าโซนิค [2]	8
2.5	การทำงานของเซนเซอร์พื่อใช้วัดระยะทาง [2]	9
2.6	ไดอะแกรมการทำงานของเซ็นเซอร์วัดระยะด้วยคลื่นอัลตร้าโซนิค [3]	10
2.7	หน้าต่างของ ARDUINO IDE	11
2.8	ESP8266 NODEMCU [4]	13
2.9	การเพิ่ม URL ลงใน ADDITION BOARD MANAGER URLS: [4]	14
2.10	การ INSTALL ESP8266 ลง ARDUINO IDE [4]	14
2.11	แสดงเมนูสำหรับเลือกใช้งาน ESP8266 ของ ARDUINO IDE [4]	15
2.12	สัญลักษณ์ในวงจรไฟฟ้าของรีเลย์ [5]	16
2.13	รีเลย์ในสาวะปกติและสภาวะจ่ายไฟ [5]	16
2.14	บอร์ด RELAY 8 ช่อง 5 โวลต์ [5]	17
2.15	โมดูล DHT 22 [6]	18
2.16	สมการคำนวณหาความชื้นสัมพัทธ์ [6]	18
2.17	สัญญาณของ MCU กับ DHT [6]	19
2.18	การส่งบิตค่า “0” ของ DHT [6]	19
2.19	การส่งบิตค่า “1” ของ DHT [6]	19
2.20	บิตข้อมูลของ DHT 22 [6]	20
2.21	เปรียบเทียบความแตกต่างระหว่าง DHT11 และ DHT22 [6]	20
2.22	ส่วนประกอบภายนอก RC SERVO MOTOR [7]	21
2.23	ส่วนประกอบภายใน RC SERVO MOTOR [7]	22
2.24	SERVO MOTOR BLOCK DIAGRAM [7]	22
2.25	มุมหรือองศากับความกว้างของสัญญาณพัลซ์ [7]	23

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.26	ความกว้างพัลซ์ที่มุม 45 องศา [7]	24
2.27	ANDROID STUDIO IDE FOR ANDROID APP [9]	26
2.28	หน้าจอหลักของโปรแกรม ANDROID STUDIO [9]	27
3.1	แผนผังการสั่งงานรีเลย์ ด้วย FIREBASE	30
3.2	แผนผังการสั่งงาน SERVO MOTOR ด้วย FIREBASE	31
3.3	แผนผังการสั่งงานรีเลย์ ด้วย FIREBASE ผ่านแอพพลิเคชั่น	32
3.4	แผนผังการสั่งงาน SERVO MOTOR ด้วย FIREBASE ผ่านแอพพลิเคชั่น	33
3.5	แผนผังการส่งค่าอุณหภูมิและความชื้นจาก NODE MCU ไปเก็บบน FIREBASE	34
3.6	แผนผังการอ่านค่าอุณหภูมิและความชื้นจาก FIREBASE มาแสดงที่แอพพลิเคชั่น	35
3.7	แผนผังการตั้งค่าการแจ้งเตือนอุณหภูมิบนแอพพลิชั่น	36
3.8	แผนผังการตั้งค่าการแจ้งเตือนอุณหภูมิบนแอพพลิชั่น	37
3.9	แผนผังแสดงแจ้งเตือนบนแอพพลิเคชั่นเมื่อได้รับคำสั่ง	38
3.10	แผนผังแสดงการอ่านค่าระดับน้ำ	39
3.11	แผนผังแสดงการอ่านค่าระดับน้ำ	40
3.12	แผนผังแสดงการแจ้งเตือนเมื่อระดับถึงค่าที่กำหนด	41
3.13	การเพิ่มโครงการของ FIREBASE	42
3.14	การใช้งาน Realtime database	42
3.15	การตั้งค่ากฎใน Realtime database	43
3.16	การเพิ่มข้อมูลลงบน Realtime database	43
3.17	การเชื่อมต่อ Android studio กับ Firebase	44
3.18	หน้า Assistant ของ Firebase	44
3.19	หน้าเชื่อมต่อ Realtime Database บน Android studio	45
3.20	หน้าต่างสำหรับเลือกการเชื่อมต่อกับ Firebase	46
3.21	หน้าต่างเชื่อต่อกับ Firebase เรียบร้อยแล้ว	47
3.22	หน้าเชื่อมต่อ Realtime database หลังจากเพิ่ม Realtime Database เรียบร้อยแล้ว	48

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1	ค่าอุณหภูมิและความชื้นบนซีเรียลมอนิเตอร์	50
4.2	สัญญาณที่ได้จากการวัดขาสัญญาณของ DHT22	51
4.3	สัญญาณ 16 ของค่าอุณหภูมิ	51
4.4	สัญญาณ 16 ของค่าความชื้น	52
4.5	ค่าอุณหภูมิและความชื้นบน REALTIME DATABASE ของ FIREBASE	53
4.6	ค่าอุณหภูมิบน SMART PHONE	54
4.7	หน้าแอพพลิเคชั่นในการควบคุมการเปิดปิด	55
4.8	ค่าที่เก็บใน FIREBASE ตอนยังไม่กดปุ่ม	56
4.9	หน้าแอพพลิเคชั่นเมื่อกดเปิด	57
4.10	ค่าที่เก็บใน FIREBASE เมื่อกดปุ่มเปิด	58
4.11	เมื่อกดปุ่มเปิดรีเลย์จะทำงาน	59
4.12	ระดับของน้ำที่จะทำการวัด	60
4.13	ระดับของน้ำใน SERIAL MONITOR จากเซนเซอร์ HC-SR04	60
4.14	ค่าเริ่มต้นของอุณหภูมิที่กำหนด	61
4.15	การกำหนดค่าอุณหภูมิในแอพพลิเคชั่น	62
4.16	ค่าที่เก็บใน FIREBASE หลังจากตั้งค่าอุณหภูมิในแอพพลิเคชั่น	63
4.17	การแจ้งเตือนเมื่ออุณหภูมิที่ตั้งค่าไว้ต่ำกว่าอุณหภูมิที่อ่านได้	64
4.18	การตั้งค่าอุณหภูมิต่ำสุดให้มีค่ามากกว่าอุณหภูมิที่อ่านได้	65
4.19	การค่าที่เก็บใน FIREBASE หลังจากการตั้งค่าอุณหภูมิ	66
4.20	การแจ้งเตือนเมื่ออุณหภูมิที่ตั้งค่าไว้มากกว่าอุณหภูมิที่อ่านได้	67
4.21	ค่าบนซีเรียลมอนิเตอร์ของ NODE MCU ESP8266	68
4.22	การค่าที่เก็บใน FIREBASE หลังจากวัดระดับน้ำ	69
4.23	การแจ้งเตือนของแอพพลิเคชั่นเมื่อระดับน้ำต่ำกว่าเกณฑ์	70

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความเป็นมาและที่มาของปัญหา

ปัจจุบันไก่พื้นบ้านได้รับความนิยมเพิ่มสูงขึ้นมาก เป็นเพราะไก่พื้นบ้านมีเนื้อรสชาติอร่อยและเนื้อแน่น เป็นที่ถูกปากของผู้บริโภคทั่วไป จนมีแนวโน้มว่า จะสามารถส่งเนื้อไก่พื้นบ้านออกไปจำหน่ายยังต่างประเทศ แต่ปัญหาคือปริมาณไก่บ้านยังไม่เพียงพอต่อความต้องการ เพราะเกษตรกรจำนวนมากประมาณร้อยละ 80 เปอร์เซ็นต์ จะเลี้ยงไก่พื้นบ้านแบบหลังบ้านประมาณ 10-20 ตัวต่อครัวเรือน ซึ่งการเลี้ยงก็เป็นการเลี้ยงแบบปล่อยตามยถากรรม จึงเป็นเหตุให้เกิดความสูญเสียพอสมควร แต่ถ้าเกษตรกรสามารถปรับใช้เทคนิคการเลี้ยงแบบเรือนโรงมาผสมผสานกับการเลี้ยงแบบพื้นบ้าน สามารถให้จำนวนไก่บ้านที่จะออกสู่ตลาดมีปริมาณที่สูงขึ้นอย่างแน่นอน จากสภาพการเลี้ยงที่ปล่อยตามธรรมชาติ และขาดการดูแลเอาใจใส่อย่างใกล้ชิด เพื่อขจัดข้อเสียของไก่พื้นบ้านดังที่กล่าวมา เนื่องจากไก่พันธุ์ดีนั้น ๆ ต้องการการดูแลเอาใจใส่อย่างสม่ำเสมอ ปริณญานิพนธ์นี้จึงได้ทำการศึกษาการทำฟาร์มสัตว์ โดยสัตว์ที่เลือกทำการศึกษาคือฟาร์มไก่เนื้อ และได้ทำการศึกษาวิธีการเลี้ยง การดูแล และได้ศึกษาโปรแกรมที่ใช้ในการทำงานเพื่อสะดวกต่อการเลี้ยงดู อีกทั้งยังป้องกันอันตรายและเฝ้าระวังภายในฟาร์มสัตว์

1.2 วัตถุประสงค์

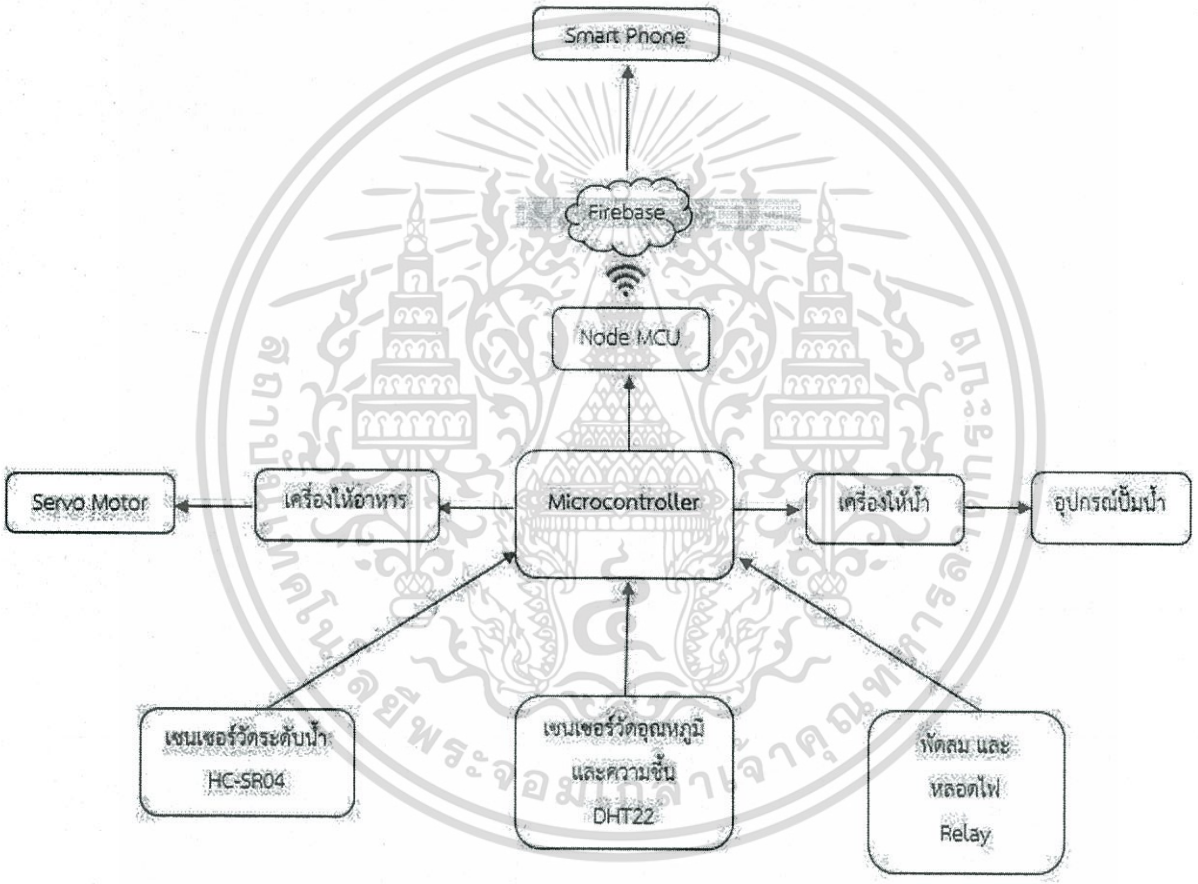
1. เพื่อศึกษาการทำฟาร์มสัตว์
2. เพื่อเพิ่มความสะดวกในการทำฟาร์มสัตว์
3. เพื่อให้สามารถตรวจสอบสถานะภายในฟาร์มสัตว์

1.3 ขอบเขตของปริญญานิพนธ์

1. สร้างระบบเฝ้าระวังภายในฟาร์มสัตว์ได้
2. ตรวจสอบสถานะภายในฟาร์มสัตว์ได้
3. ใช้แอพพลิเคชั่นในสมาร์ทโฟนควบคุมระบบได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บล็อกไดอะแกรมของปริญญานิพนธ์



รูปที่ 1.1 บล็อกไดอะแกรมของปริญญานิพนธ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

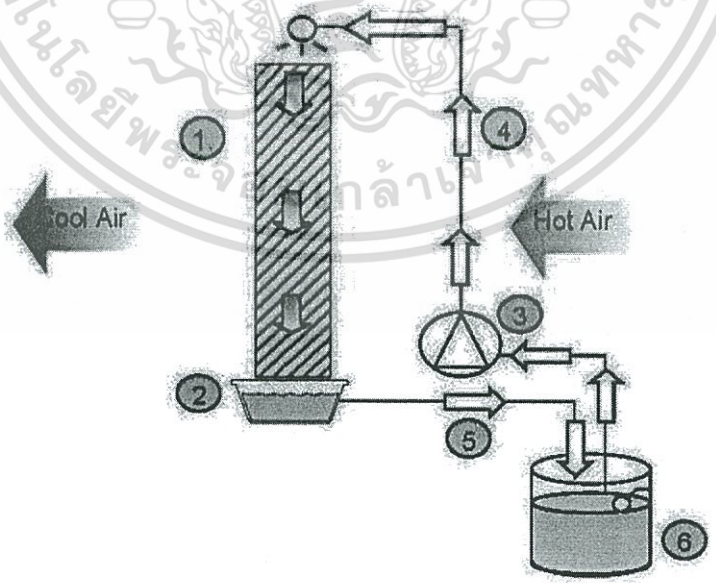
ทฤษฎีและหลักการที่เกี่ยวข้อง

2.1 ความรู้เกี่ยวกับโรงเรือนระบบฟาร์มปิด

2.1.1 ความหมายของระบบฟาร์มปิด

ระบบ EVAP (Evaporative Air Cooling System)นำมาทดแทน ระบบแอร์ ซึ่งจะช่วยประหยัดเงินในการติดตั้ง ระบบแอร์ใหม่ได้หลาย เท่า และ ยังช่วยประหยัดค่าไฟ ได้อีก ไม่น้อยกว่า 80% สามารถ ควบคุม เรื่องความชื้น ให้เหมาะกับ ความต้องการ ในรูปแบบ การใช้งาน ที่ต่างๆ ได้

ในสภาวะเศรษฐกิจปัจจุบันที่แนวโน้มค่าไฟฟ้าเริ่มสูงขึ้น ระบบปรับอากาศที่ใช้พลังงานธรรมชาติจะช่วยให้เจ้าของกิจการประหยัดไฟ 80% การใช้ EVAP หมายถึง ระบบทำความเย็นที่ใช้พลังงานธรรมชาติ ใช้การระเหยของน้ำช่วยในการทำความเย็น หลักการทำงานของ EVAP ก็คือ ปล่อยกระแสลม ไหลผ่านตัวกลางที่มีน้ำไหลผ่าน (cooling pad) และการที่อากาศซึ่งมีอุณหภูมิสูงกว่าไหลผ่านน้ำจะทำให้น้ำระเหยกลายเป็นไอ ซึ่งจะเป็นการดึงเอาความร้อนของอากาศออก ทำให้อุณหภูมิของอากาศลดลง

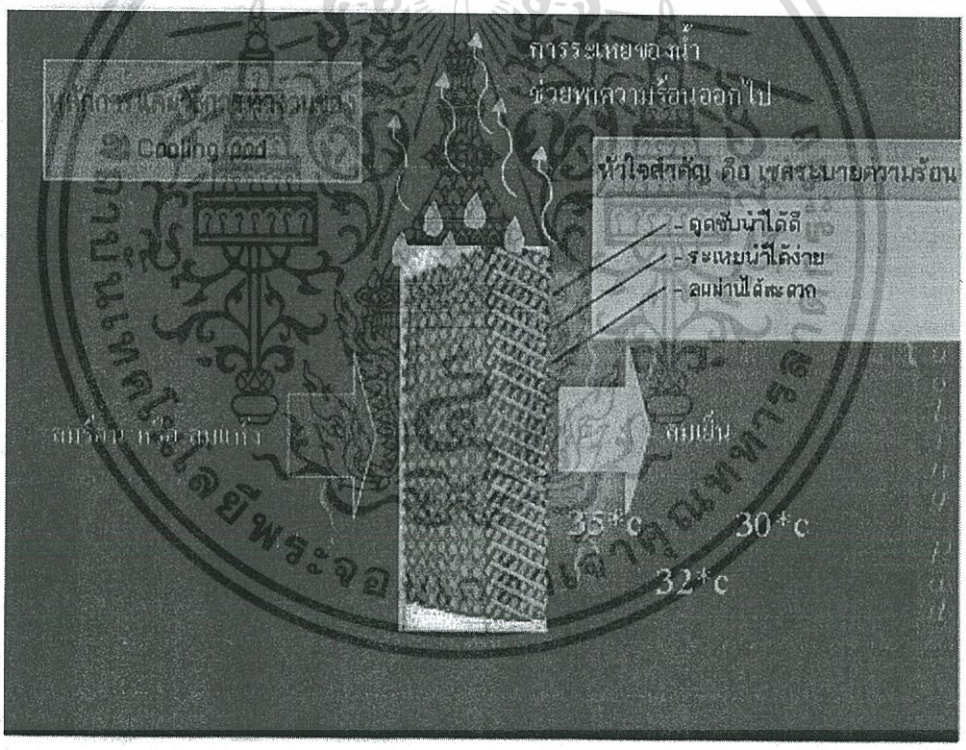


รูปที่ 2.1 การทำงานของระบบ Evap [1]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.2 หลักการทำงานของระบบ Evap

หลักการทำงานของระบบ อีแว็ป (Evap) คือ ปล่อยกระแสลม ไหลผ่านตัวกลางที่มีน้ำไหลผ่าน คลูลิ่งแพด (Cooling pad) และการที่อากาศซึ่งมีอุณหภูมิสูงกว่าไหลผ่านน้ำจะทำให้น้ำระเหยกลายเป็นไอ ซึ่งจะเป็นการดึงเอาความร้อนของอากาศออกทำให้อุณหภูมิของอากาศลดลง จากหลักการนี้ทำให้มีการนำมาออกแบบใช้กับการทำโรงเรือนเลี้ยงสัตว์หรือปลูกพืชได้ โดยมีการติดตั้งแผงความเย็นคลูลิ่งแพด (Cooling pad) ที่ทำจากกระดาษอัดเคลือบน้ำยาชนิดพิเศษ ซึ่งกระดาษอัดชนิดน้ำได้มีการนำมาเรียงเป็นแท่งมีความหนาประมาณ 4-6 นิ้ว และวางเรียงให้มีพื้นที่ผิวมากที่สุด เพื่อให้มีโอกาสสัมผัสน้ำและอากาศที่ไหลผ่านมากที่สุด จะเป็นการช่วยให้น้ำมีการระเหยได้มากนั่นเอง และมีพัดลมดูดเอาอากาศร้อนข้างนอกผ่านแผง คลูลิ่งแพด (Cooling pad) ที่มีน้ำไหลผ่าน



รูปที่ 2.2 หลักการทำและวิธีการทำงานของ Cooling pad [1]

คูลลิ่งแพด (Cooling pad) เป็นหัวใจของระบบ อีแว็ป (Evap) ผลิตจากวัสดุที่เป็นเยื่อไม้จากธรรมชาติเคลือบสารชนิดพิเศษที่ดูดซับน้ำได้รวดเร็ว และไม่เปื่อยยุ่ย สามารถกระจายน้ำไปทั่วทุก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผื้นผิวได้อย่างรวดเร็วและสม่ำเสมอ และยังออกแบบร่องอากาศให้มีความลาดเอียง เพื่อช่วยให้น้ำที่เหลือจากการระเหยไหลลงมาชำระล้างฝุ่นและสิ่งสกปรกออกจากพื้นผิว จึงช่วยเพิ่มประสิทธิภาพในการกรองฝุ่นละอองที่ปะปนในอากาศภายนอก ระบบจะทำงานได้เต็มประสิทธิภาพต้องดูแลรักษาความสะอาดอย่างสม่ำเสมอ เหตุผลที่ทำให้น้ำสามารถระเหยกลายเป็นไอได้นี้มาจากการที่ความชื้นสัมพัทธ์ในอากาศต่ำกว่าความชื้นของน้ำ (100%) ทำให้น้ำสามารถระเหยได้ ดังนั้นน้ำจะระเหยได้มากเพียงใดก็จะขึ้นอยู่กับความชื้นสัมพัทธ์ของอากาศ ในขณะนั้น ยิ่งอากาศมีความชื้นต่ำมากเพียงใด ความสามารถในการระเหยก็จะมากขึ้นเท่านั้นและก็จะทำให้อุณหภูมิลดต่ำลงได้มากเท่านั้นเช่นกัน

ระบบอีแว๊ป (Evap) สามารถใช้ได้กับ ศูนย์สรรพสินค้า ศูนย์กีฬาในร่ม งานแสดงสินค้า นิทรรศการ รวมทั้งอุตสาหกรรมการเกษตร เช่น โรงเรือนปศุสัตว์ และโรงเรือนเพาะชำ หากเทียบกับเครื่องปรับอากาศจะใช้เงินลงทุนและค่าดูแลรักษาต่ำกว่า อีกทั้งยังประหยัดค่าไฟได้มากกว่า 80% หรือใช้พลังงานไฟฟ้าเพียง 20% และจุดคุ้มทุนเพียง 1 ปี หากเป็นอุตสาหกรรมที่มีฝุ่นในพื้นที่ ผลิตมาก ระบบนี้ยังช่วยลดปริมาณฝุ่นได้อีกด้วย เนื่องจากในระบบมีอากาศถ่ายเทตลอดเวลาเพียงแต่ผู้เชี่ยวชาญจะต้องสำรวจอาคารและองค์ประกอบต่างๆ เช่นทิศทางของอาคาร ทิศทางของลมภายนอก การวางตำแหน่งของเครื่องจักรที่ผลิต ตำแหน่งของเครื่องจักรที่ให้ความร้อน เพื่อให้ได้ระบบที่เหมาะสมกับอาคารนั้นๆ

จากหลักการอันนี้ทำให้มีการนำมาออกแบบใช้กับการทำโรงเรือนเลี้ยงสัตว์หรือปลูกพืชได้ โดยมีการติดตั้งแผงความเย็น (cooling pad) ที่ทำจากกระดาษอัดเคลือบน้ำยาชนิดพิเศษ ซึ่งกระดาษอัดชนิดน้ำได้มีการนำมาเรียงเป็นแท่งมีความหนาประมาณ 4-6 นิ้ว และวางเรียงให้มีพื้นที่ผิวมากที่สุดเพื่อให้มีโอกาสสัมผัสน้ำและอากาศที่ ไหลผ่านมากที่สุด อันจะเป็นการช่วยให้น้ำมีการระเหยได้มากนั่นเอง และมีพัดลมดูดเอาอากาศร้อนข้างนอกผ่านแผง cooling pad

โรงเรือน EVAP เป็นระบบฟาร์มปิด สามารถความคุมอุณหภูมิและความชื้นให้คงที่ตามที่ตั้งไว้ได้ และยังสามารถป้องกันแมลงและสามารถควบคุมเรื่องความชื้น ให้เหมาะกับสิ่งต่างๆ ได้ ข้อดีของระบบฟาร์มปิด คือการควบคุมอุณหภูมิให้อยู่ในระดับที่เหมาะสมและป้องกันโรคระบาด ระบบระบบฟาร์ม ปิด หรือ อีแวป นับเป็นอีกก้าวในการพัฒนากระบวนการผลิตอาหารป้อนครัวโลก โดยคำนึงถึงคุณภาพและความปลอดภัยของผู้บริโภคเป็นสำคัญ สามารถให้ผลผลิตได้ตลอดทั้งปี เกิด ประโยชน์ สูงสุด ต่อเจ้าของ และ เป็นมิตรกับสิ่งแวดล้อม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.3 ข้อดีของระบบ Evap

ข้อดีของการทำโรงเรือนเลี้ยงสัตว์ในระบบ EVAP

1. สามารถป้องกันโรคระบาดได้อย่างมีประสิทธิภาพ

2. สามารถเพิ่มจำนวนไก่ต่อหนึ่งหน่วยพื้นที่ โดยสามารถเลี้ยงไก่ได้ประมาณ 15 ตัวต่อตารางเมตร (จากปกติ 8-9 ตัวต่อตารางเมตร)

3. อัตราการเติบโตดีขึ้น

4. น้ำหนักเฉลี่ยต่อตัวสูงกว่าการเลี้ยงในโรงเรือนแบบเปิด ในระยะเวลาเท่ากัน

5. อัตราการตายต่ำ

6. สามารถป้องกันแมลงและศัตรูที่มารบกวนได้

7. สามารถจัดการให้แสงสว่างภายในโรงเรือนได้อย่างมีประสิทธิภาพ

8. ลดความชื้นและก๊าซแอมโมเนียภายในโรงเรือนลงได้

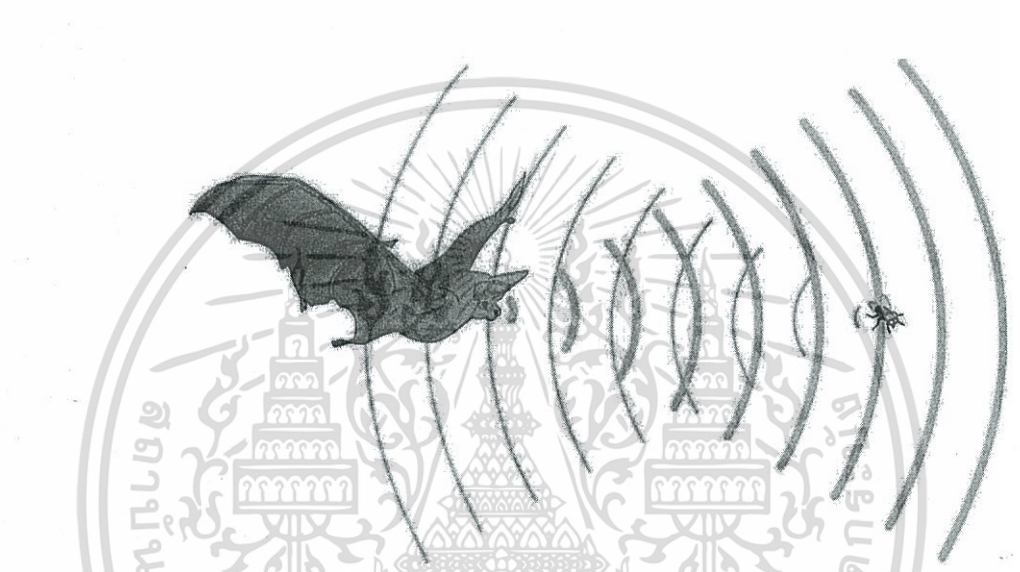
9. เป็นการใช้แรงงานอย่างมีประสิทธิภาพ โดยเฉพาะในโรงเรือนที่มีการให้น้ำและอาหารแบบอัตโนมัติ

โรงเรือนเลี้ยงสัตว์จัดเป็นปัจจัยหนึ่งที่มีความสำคัญต่อวงจรการเลี้ยงสัตว์ทุกชนิด ทั้งนี้เพราะโรงเรือนเปรียบเสมือนที่อยู่ ที่หลับนอนของสัตว์ หากผู้เลี้ยงสัตว์สามารถจัดหาโรงเรือนที่เหมาะสม มีสุขลักษณะที่ดีให้กับสัตว์แล้วสัตว์ก็จะให้ผลตอบแทนอย่างสูงสุด การสร้างโรงเรือนก็ต้องสร้างให้เหมาะสมกับชนิด พันธุ์ เพศ และอายุของสัตว์ เนื่องจากสัตว์ในแต่ละระยะต้องการโรงเรือนที่ต่างกันแต่สิ่งหนึ่งที่สัตว์ทุกชนิดต้องการเหมือนกันคือโรงเรือนที่เย็นสบายอากาศถ่ายเทได้สะดวก แห้งและไม่เปียกชื้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 คลื่นอัลตร้าโซนิค (Ultra-sonic wave)

Ultrasonic หมายถึง คลื่นเสียงที่มีความถี่สูงเกินกว่าที่หูมนุษย์จะได้ยิน โดยทั่วไปแล้วหูของมนุษย์โดยเฉลี่ยจะได้ยินเสียงสูงถึงเพียงแค่ประมาณ 15 kHz เท่านั้น แต่พวกที่อายุยังน้อย ๆ อาจจะได้ยินเสียงที่มีความถี่สูงกว่านี้ได้ ดังนั้นโดยปกติแล้วคำว่าอัลตร้าโซนิคจึงมักจะหมายถึงคลื่นเสียงที่มีความถี่สูงกว่า 20 kHz ขึ้นไป จะสูงขึ้นจนถึงเท่าใดไม่ได้ระบุจำกัดเอาไว้



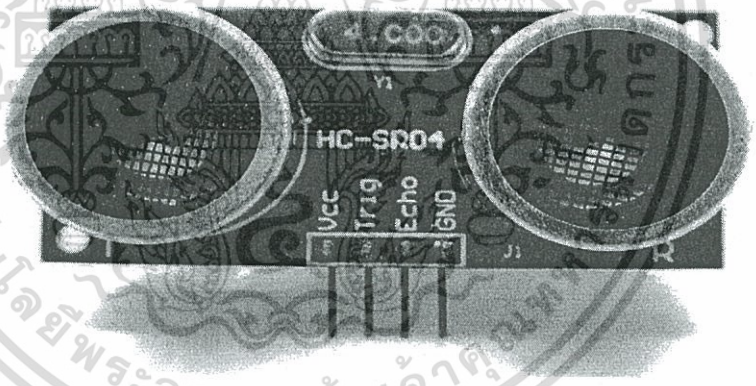
รูปที่ 2.3 คลื่นอัลตร้าโซนิค [2]

จากรูปที่ 2.3 การใช้คลื่นอัลต้าโซนิคของค้างคาว สาเหตุที่มีการนำเอาคลื่นย่านอัลตร้าโซนิคมาใช้ก็เพราะว่าเป็นคลื่นที่มีทิศทางทำให้เราสามารถเล็งคลื่นเสียงไปยังเป้าหมายที่ต้องการได้โดยเจาะจง เรื่องนี้เป็นคุณสมบัติของคลื่นอย่างหนึ่ง ยิ่งคลื่นมีความถี่สูงขึ้นความยาวคลื่นก็จะยิ่งสั้นลง ถ้าความยาวคลื่นยาวกว่าช่องเปิดของตัวกำเนิดเสียงความถี่นั้นเช่น คลื่นความถี่ 300 Hz ในอากาศจะมีความยาวถึงประมาณ 1 เมตรเศษ ๆ ซึ่งจะยาวกว่าช่องที่ให้คลื่นเสียงออกมาจากตัวกำเนิดเสียงโดยทั่วไปมากมายคลื่นจะหักเบนที่ขอบด้านนอกของตัวกำเนิดเสียงทำให้เกิดการกระจายทิศทางคลื่นแต่ถ้าความถี่สูงขึ้นมาอยู่ในย่านอัลตร้าโซนิค อย่างเช่น 40 kHz จะมีความยาวคลื่นในอากาศเพียงประมาณ 8 มม. เท่านั้นซึ่งเล็กกว่ารูเปิดของตัวที่ให้กำเนิดเสียงความถี่นี้มาก คลื่นเสียงจะไม่มีการเลี้ยวเบนที่ขอบจึงพุ่งออกมาเป็นลำแคบๆหรือที่เราเรียกว่า“มีทิศทาง” การมี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทิศทางของคลื่นเสียงย่านอัลตร้าโซนิคทำให้เรานำไปใช้งานได้หลายอย่าง เช่น นำไปใช้ในเครื่องควบคุมระยะไกล (Ultrasonic remote control) เครื่องล้างอุปกรณ์ (Ultrasonic cleaner) โดยให้น้ำสั่นที่ความถี่สูง เครื่องวัดความหนาของวัตถุโดยสังเกตระยะเวลาที่คลื่นสะท้อนกลับมา เครื่องวัดความลึกและทำแผนที่ใต้ท้องทะเล ใช้ในเครื่องหาตำแหน่งอวัยวะบางส่วนในร่างกาย ใช้ทดสอบการรั่วไหลของท่อ เป็นต้น โดยความถี่ที่ใช้ขึ้นอยู่กับการใช้งาน เช่น คลื่นเสียงต้องเดินทางผ่านอากาศแล้ว ความถี่ที่ใช้ก็มักจะจำกัดอยู่เพียงไม่เกิน 50 kHz เพราะที่ความถี่สูงขึ้นกว่านี้อากาศจะดูดกลืนคลื่นเสียงเพิ่มขึ้นมาก ทำให้ระดับความแรงของคลื่นเสียงที่ระยะห่างออกไปลดลงอย่างรวดเร็ว ส่วนการใช้งานด้านการแพทย์ซึ่งต้องการรัศมีทำการสั้น ๆ ก็อาจใช้ความถี่ในช่วง 1 MHz ถึง 10 MHz ขณะที่ความถี่เป็น GHz ก็มีใช้กันในหลายๆ การใช้งานที่ตัวกลางที่คลื่นเสียงเดินทางผ่านไม่ใช่อากาศ

2.2.2 เซนเซอร์อัลตร้าโซนิค และ การใช้วัดระยะทาง

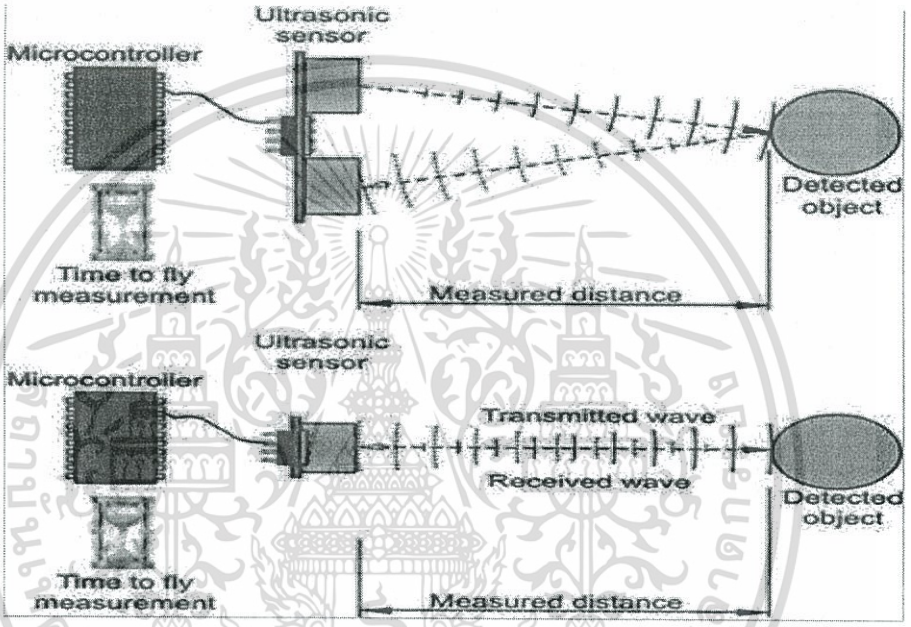


รูปที่ 2.4 เซนเซอร์อัลตร้าโซนิค [2]

รูปที่ 2.4 เป็นเซนเซอร์วัดระยะทางโดยใช้คลื่นอัลตร้าโซนิค รุ่น SRF04 เป็นแผงวงจรตรวจจับและ วัดระยะทางด้วยคลื่นอัลตร้าโซนิกที่มีความเที่ยงตรงสูง โดยสามารถวัดระยะได้ตั้งแต่ 1 เซนติเมตรจนถึง 4 เมตร โมดูล SRF-04 ให้ผลลัพธ์เป็นค่าความ กว้างพัลส์แปรผันกับระยะทางที่ตรวจวัดได้ออกแบบมาให้ต่อใช้งานกับไมโครคอนโทรลเลอร์ได้ง่าย โดยใช้สายสัญญาณเพียง 2 เส้น โมดูล HC-SR04 ทำงานที่แรงดันประมาณ +5V (4.5V ถึง +5.5V) โดยป้อนให้ขา VCC และ GND

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

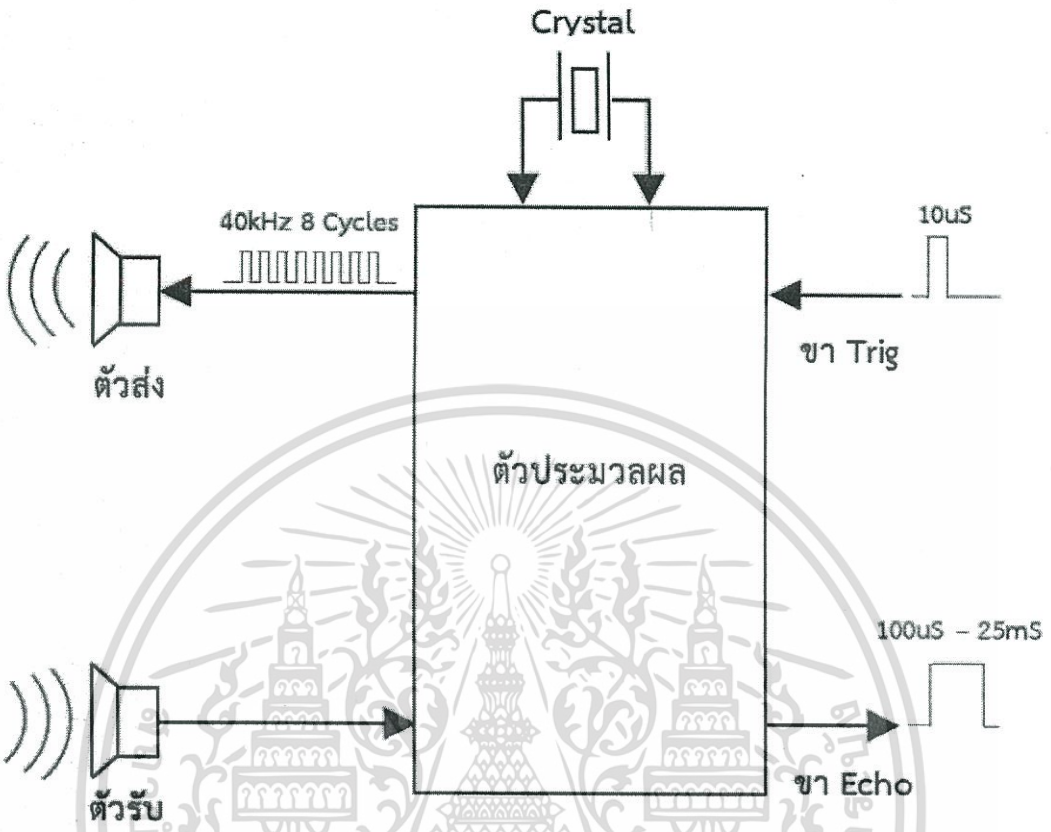
โมดูลนี้ มีขาสัญญาณดิจิทัล TRIG (อินพุต) และ ECHO (เอาต์พุต) ที่นำไปเชื่อมต่อกับไมโครคอนโทรลเลอร์ได้ อย่างเช่น Arduino ในการวัดระยะห่างแต่ละครั้ง จะต้องสร้างสัญญาณแบบ Pulse ที่มีความกว้าง (Pulse Width) อย่างน้อย 10 us ป้อนให้ขา TRIG และหลังจากนั้นให้วัดความกว้างของสัญญาณช่วง HIGH จากขา ECHO ถ้าวัตถุอยู่ใกล้ ความกว้างของสัญญาณ Pulse ที่ได้ก็จะน้อย แต่ถ้าวัตถุอยู่ไกลออกไป ก็จะได้ค่าความกว้างของสัญญาณ Pulse ที่มากขึ้น



รูปที่ 2.5 การทำงานของเซนเซอร์เพื่อใช้วัดระยะทาง [2]

รูปที่ 2.5 แสดงการทำงานของเซนเซอร์เพื่อใช้วัดระยะทางเริ่มต้นจะต้องให้สัญญาณขา Trig มีสถานะทางลอจิกเป็น LOW เสียก่อน จากนั้นจึงเริ่มทริกสัญญาณ โดยให้ขา Trig มีสถานะเป็น HIGH ค้างไว้อย่างน้อย 10us แล้วจึงปรับสถานะเป็น LOW จากนั้น ที่ขา Echo ให้เตรียมรับสัญญาณทริก HIGH กลับมาเมื่อมีการส่งสัญญาณ HIGH กลับมา ให้เริ่มนับเวลาที่สัญญาณเป็น HIGH และเมื่อสัญญาณขา Echo กลับเป็น LOW ให้สิ้นสุดการนับเวลา แล้วจึงนำค่าเวลาที่นับได้ไปคำนวณอีกที ซึ่งในการคำนวณนั้น จะขึ้นอยู่กับรุ่น ในบางรุ่นสามารถใช้ค่าอัตราเร็วเสียงมาคำนวณได้เลย แต่ในบางรุ่น ต้องใช้สูตรคำนวณเฉพาะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



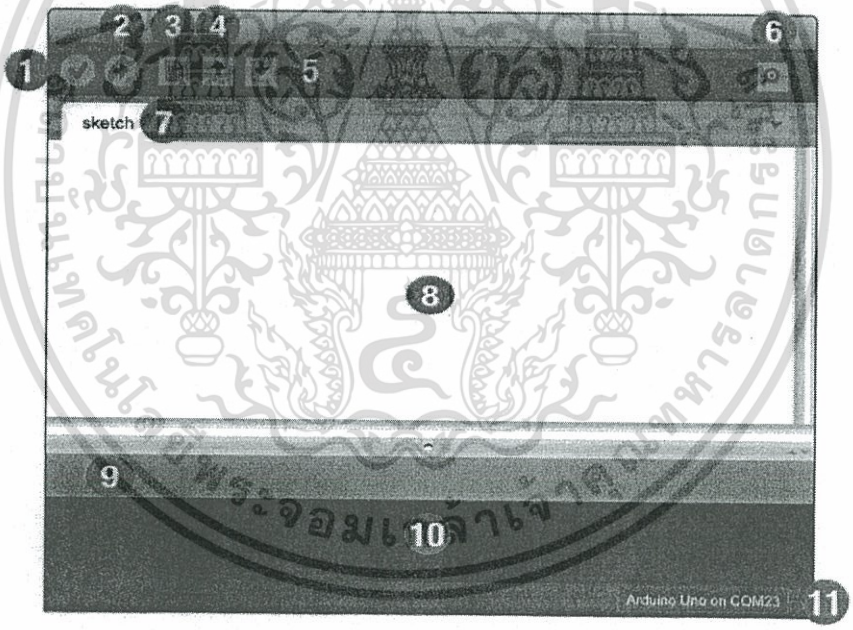
รูปที่ 2.6 การทำงานของเซ็นเซอร์วัดระยะด้วยคลื่นอัลตร้าโซนิค [3]

จากรูปที่ 2.6 แสดงการส่งสัญญาณเข้าไปที่ Trig วงจรภายในจะเริ่มสร้างความถี่ 40 kHz จำนวน 8 ลูกคลื่นออกไป โดยใช้ความถี่จากคริสตัลเป็นตัวอ้างอิง แล้วตัวส่งที่เปรียบเสมือนลำโพง จะส่งสัญญาณออกไป จากนั้นเมื่อคลื่นวิ่งกลับมาที่ตัวรับ ที่เปรียบเสมือนเป็นไมโครโฟน สัญญาณไฟฟ้าจะผ่านตัวประมวลผล แล้วให้ค่าเอาต์พุตออกมาทางขา Echo จะเห็นว่า แกนหลักของเซ็นเซอร์จะเป็นตัวประมวลผล ซึ่งตัวประมวลผลนี้ ในแต่ละรุ่นก็จะแตกต่างกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 Arduino IDE

Arduino IDE (Arduino Integrated Development Environment) เครื่องมือการเขียนโปรแกรมที่มีใช้งานได้กับอาดูโน่ได้ทุกรุ่น โดยภายในจะมีเครื่องมือที่จะเป็นสำหรับติดต่ออาดูโน่ เช่น การค้นหาอาดูโน่ ที่ติดต่อกับเครื่องคอมพิวเตอร์ การเลือกรุ่นอาดูโน่ที่ต่ออยู่เพื่อตรวจสอบว่าขนาดของโปรแกรมที่เขียน หรือไรบรารี่ต่างๆซับพอร์ตกับอาดูโน่รุ่นนั้นๆหรือไม่ อีกทั้งยังมีโปรแกรมติดต่อผ่านซีเรียลโดยตรงสำหรับคอมพิวเตอร์โปรแกรมอาดูโน่ไอดีอี เป็นโปรแกรมโอเพ่นซอสสามารถนำไปใช้งานได้ทั่วไปสำหรับบอร์ด Arduino นั่นคือโปรแกรมที่เรียกว่า Arduino IDE ในการเขียนโปรแกรมและคอมไพล์ลงบอร์ด โดย ขนาดของโปรแกรม Arduino โดยปกติแล้วจะใหญ่กว่าโค้ด AVR ปกติเนื่องจากโค้ด AVR เป็นการเข้าถึงจาก รีจิสเตอร์โดยตรง แต่โค้ด Arduino เข้าถึงผ่านฟังก์ชั่น เพื่อให้สามารถเขียนโค้ดได้ง่าย



รูปที่ 2.7 หน้าต่างของ Arduino IDE [2]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.7 หน้าต่างของArduino IDE ซึ่งมีส่วนประกอบดังนี้

1.) Verify: คอมไพล์และอนุมัติโปรแกรมที่เราเขียนไป จะตรวจสอบข้อผิดพลาดในไวยากรณ์

2.) Upload: อัพโหลดโค้ดไปยังบอร์ด.

3.) New: ปุ่มนี้จะเปิดแท็บหน้าต่างใหม่

4.) Open: ปุ่มนี้จะช่วยให้เปิดสเก็ตช์ที่มีอยู่.

5.) Save: บันทึกโปรแกรม.

6.) Serial Monitor: คลิกปุ่มนี้เมื่อต้อง การเปิดหน้าต่างสื่อสำรองและแสดงข้อมูลอนุกรม โดยต้องมีการต่อฮาร์ดแวร์ Arduino และเลือกพอร์ตการเชื่อม ต่อให้ถูกต้องก่อน

7.) Sketch Name: แสดงแบบร่างโปรแกรมที่กำลังทำการเขียนอยู่.

8.) Code Area: เป็นพื้นที่สำหรับเขียนโปรแกรมภาษา C/C++

9.) Message Area: บอกว่ามีข้อผิดพลาดในโค้ด

10.) Text Console: คอนโซลข้อความแสดงข้อความแสดงข้อผิดพลาด เมื่อเกิดจุดบกพร่องคอนโซลข้อความจะมีประโยชน์มากในการแก้ไขตัวโปรแกรม

11.) Board and Serial Port: แสดงชื่อบอร์ดและพอร์ตอนุกรมที่เลือก.

2.4 ไมโครคอนโทรลเลอร์

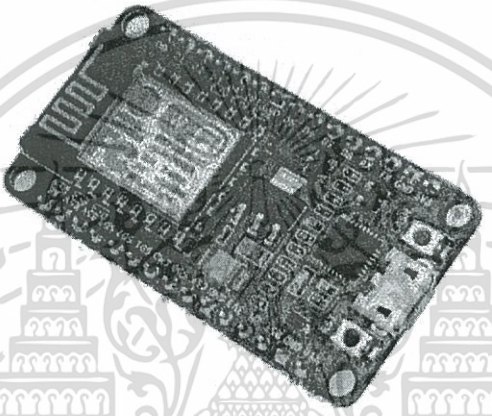
ไมโครคอนโทรลเลอร์ คือ อุปกรณ์ควบคุมขนาดเล็กซึ่ง บรรจุความสามารถที่คล้ายกับระบบคอมพิวเตอร์โดยในไมโครคอนโทรลเลอร์ได้รวมเอาซีพียู, หน่วยความจำ และพอร์ต ซึ่งเป็นส่วนประกอบหลักสำคัญของระบบคอมพิวเตอร์เข้าไว้ด้วยกัน โดยทำการบรรจุเข้าไว้ในตัวถังเดียวกัน

2.4.1 ไมโครคอนโทรลเลอร์ ESP8266 NodeMCU

NodeMCU คือ แพลตฟอร์มหนึ่งที่ใช้ช่วยในการสร้างโปรเจค Internet of Things (IoT) ที่ประกอบไปด้วย Development Kit (ตัวบอร์ด) และ Firmware (Software บนบอร์ด) ที่เป็น open source สามารถเขียนโปรแกรมด้วยภาษา Lau ได้ ทำให้ใช้งานได้ง่ายขึ้น มาพร้อมกับโมดูล WiFi (ESP8266) ซึ่งเป็นหัวใจสำคัญในการใช้เชื่อมต่อกับอินเตอร์เน็ตนั่นเอง ตัวโมดูล ESP8266นั้นมีอยู่ด้วยกันหลายรุ่นมาก ตั้งแต่เวอร์ชันแรกที่เป็น ESP-01 ไล่ไปเรื่อยๆจนปัจจุบันมีถึง ESP-12 แล้ว และที่ฝังอยู่ในNodeMCU version แรกนั้นก็เป็น ESP-12 แต่ใน version2 นั้นจะใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เป็น ESP-12E แทน ซึ่งการใช้งานโดยรวมก็ไม่แตกต่างกันมากนัก NodeMCUนั้นมีลักษณะคล้ายกับ Arduino ตรงที่มีพอร์ต Input Output buil inมาในตัว สามารถเขียนโปรแกรมคอนโทรลอุปกรณ์ I/O ได้โดยไม่ต้องผ่านอุปกรณ์อื่นๆ และเมื่อไม่นานมานี้ก็มีนักพัฒนาที่สามารถทำให้ Arduino IDE ใช้งานร่วมกับ Node MCU ได้ จึงทำให้ใช้ภาษา C/C++ ในการเขียนโปรแกรมได้ ทำให้เราสามารถใช้งานมันได้หลากหลายมากยิ่งขึ้น NodeMCUตัวนี้สามารถทำอะไรได้หลายอย่างมากโดยเฉพาะเรื่องที่เกี่ยวข้องกับ IoT ไม่ว่าจะเป็นการทำ Web Server ขนาดเล็ก การควบคุมการเปิดปิดไฟผ่านWiFi และอื่นๆอีกมากมาย



รูปที่ 2.8 ESP8266 NodeMCU [4]

การติดตั้ง Arduino IDE ลงบน ESP8266 NodeMCU

ติดตั้ง Arduino IDE เวอร์ชั้น 1.6.4 หรือ ใหม่กว่า โดย Download ตัวติดตั้งได้จาก

http://www.arduino.cc/en/main/software

เมื่อทำการติดตั้ง Arduino IDE เรียบร้อยแล้ว ให้เปิด Arduino IDE ขึ้นมา

ไปที่ Menu File >> Preferences

ใส่ URL >> ลงใน Addition Board Manager URLs:

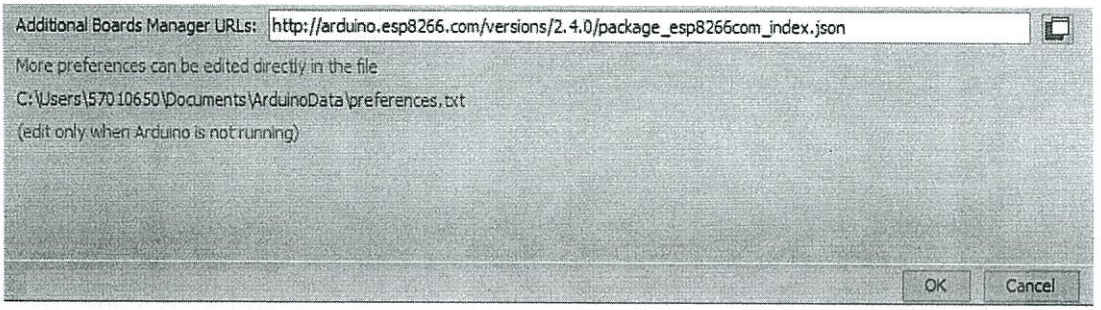
ดังนี้ http://arduino.esp8266.com/stable/package_esp8266com_index.json

(สามารถดูข้อมูลเพิ่มเติมหากมีการเปลี่ยนแปลงได้

จาก https://github.com/esp8266/Arduino)

แล้วกด OK

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

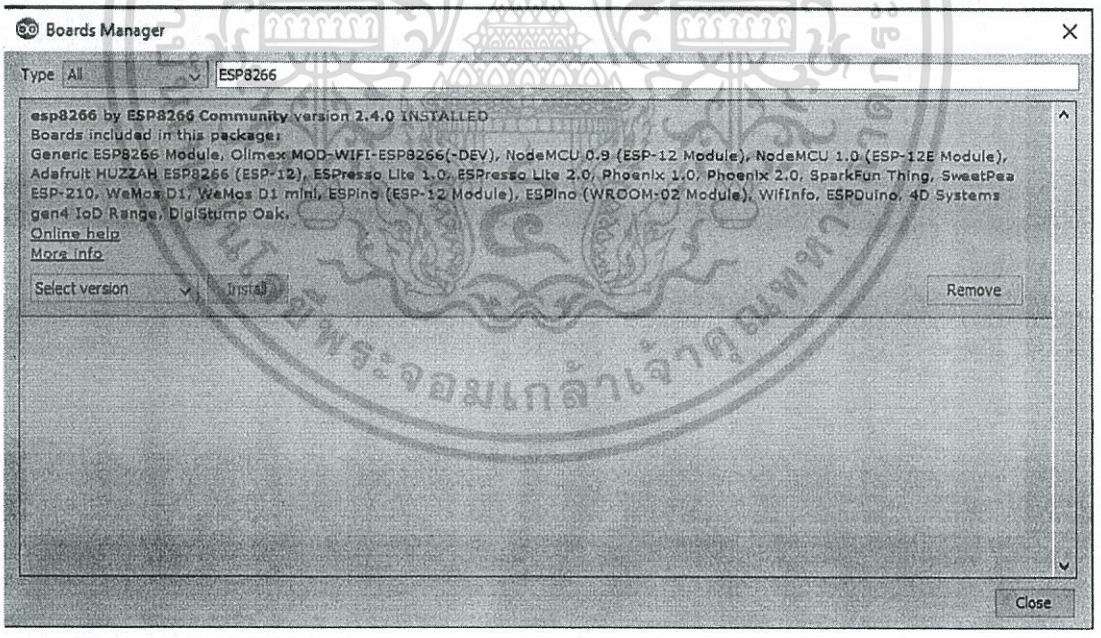


รูปที่ 2.9 การเพิ่ม URL ลงใน Addition Board Manager URLs: [4]

จากนั้นไปที่ Menu Tools >> Boar:"xxxxxx" >> Board Manager...

เลือก Type เป็น Contributed ไปที่ ESP8266 และกด Install

รอจนติดตั้งเสร็จ

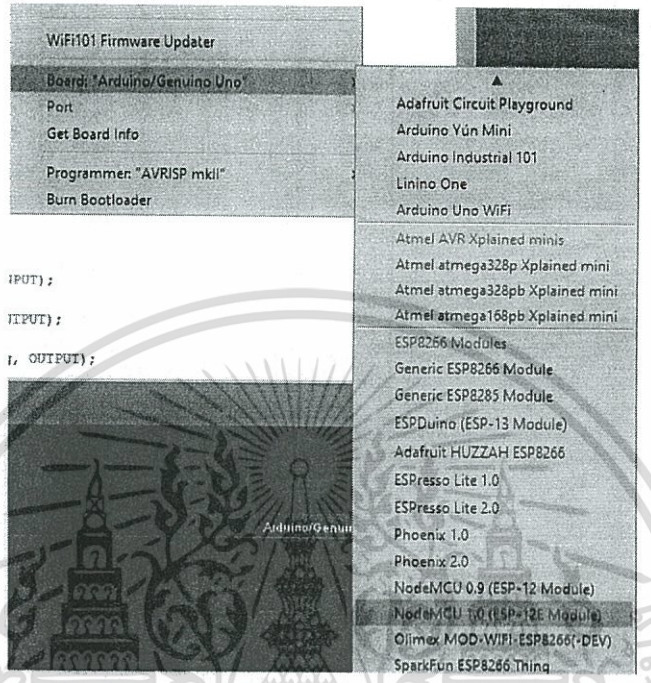


รูปที่ 2.10 การ Install ESP8266 ลง Arduino IDE [4]

เมื่อติดตั้ง ESP8266 เสร็จเรียบร้อยแล้ว ให้ปิดโปรแกรม Arduino IDE ก่อน แล้วจึงเปิดขึ้นมาใหม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อเปิดโปรแกรม Arduino IDE ขึ้นมาใหม่ ให้ลองไปที่ Menu Tools >> Board:"xxxxxx" จะพบว่า มี Menu สำหรับเลือกใช้งาน ESP8266 กับ Arduino IDE ขึ้นมาให้เลือกใช้งานแล้ว



รูปที่ 2.11 แสดงเมนูสำหรับเลือกใช้งาน ESP8266 ของ Arduino IDE [4]

ESP8266 จะมีบอร์ดให้เลือกใช้งานอยู่ด้วยกัน 3 บอร์ด ได้แก่

o Generic ESP8266 Module >> บอร์ด ESP8266 ทั้งไปไม่เจาะจง หรือ บอร์ดที่สร้างขึ้นเอง

o Olimex MOD-WIFI-ESP8266 >> บอร์ด ESP8366 ที่บริษัท Olimex เป็นผู้สร้าง

o NodeMCU (ESP8266 ESP12) >> บอร์ด ESP8266 ที่เป็นบอร์ด NodeMCU ที่เราจะใช้งานนั่นเอง

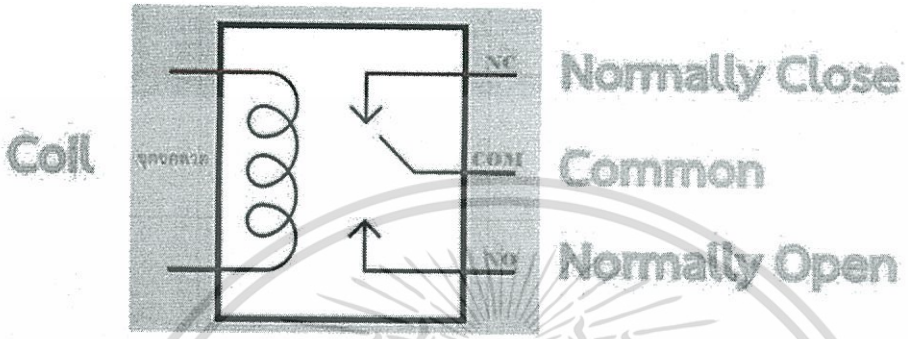
2.5 Relay switch

รีเลย์ (Relay) เป็นอุปกรณ์ไฟฟ้าชนิดหนึ่ง ซึ่งทำหน้าที่ตัดต่อวงจรแบบเดียวกับสวิตช์ โดยควบคุมการทำงานด้วยไฟฟ้า Relay มีหลายประเภท ตั้งแต่ Relay ขนาดเล็กที่ใช้ในงานอิเล็กทรอนิกส์ทั่วไป จนถึง Relay ขนาดใหญ่ที่ใช้ในงานไฟฟ้าแรงสูง โดยมีรูปร่างหน้าตาแตกต่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กันออกไป แต่มีหลักการทำงานที่คล้ายคลึงกัน สำหรับการนำ Relay ไปใช้งาน จะใช้ในการตัดต่อวงจร ทั้งนี้ Relay ยังสามารถเลือกใช้งานได้หลากหลายรูปแบบ

2.5.1 การทำงานของรีเลย์สวิตซ์



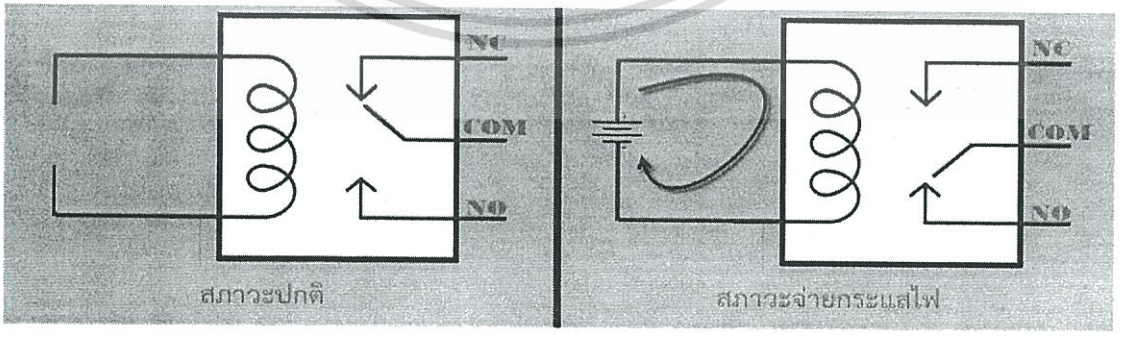
รูปที่ 2.12 สัญลักษณ์ในวงจรไฟฟ้าของรีเลย์ [5]

จากรูปที่ 2.12 ภายใน Relay จะประกอบไปด้วยขดลวดและหน้าสัมผัส

- หน้าสัมผัส NC (Normally Close) เป็นหน้าสัมผัสปกติปิด โดยในสภาวะปกติหน้าสัมผัสนี้จะต่อเข้ากับขา COM (Common) และจะลอยหรือไม่สัมผัสกันเมื่อมีกระแสไฟฟ้าไหลผ่านขดลวด

- หน้าสัมผัส NO (Normally Open) เป็นหน้าสัมผัสปกติเปิด โดยในสภาวะปกติจะลอยอยู่ ไม่ถูกต่อกับขา COM (Common) แต่จะเชื่อมต่อกันเมื่อมีกระแสไฟไหลผ่านขดลวด

- ขา COM (Common) เป็นขาที่ถูกใช้งานร่วมกันระหว่าง NC และ NO ขึ้นอยู่กับว่าขณะนั้นมีกระแสไฟฟ้าไหลผ่านขดลวดหรือไม่

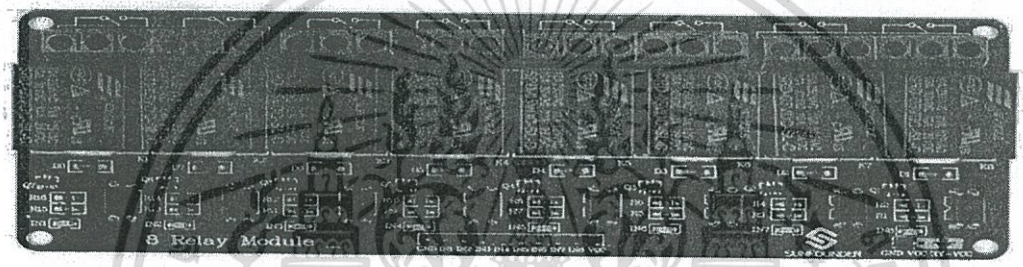


รูปที่ 2.13 รีเลย์ในสภาวะปกติและสภาวะจ่ายไฟ [5]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 2.13 เราจะใช้งาน Relay แบบ SPDT (Single Pole Double Throw) หลักการทำงานของ Relay นั้น ในส่วนของขดลวด เมื่อมีกระแสไฟฟ้าไหลผ่าน จะทำให้ขดลวดเกิดการเหนี่ยวนำและทำหน้าที่เสมือนแม่เหล็กไฟฟ้า ส่งผลให้ขา COM ที่เชื่อมต่ออยู่กับหน้าสัมผัส NC (ในสภาวะที่ยังไม่เกิดการเหนี่ยวนำ) ย้ายกลับเชื่อมต่อกับหน้าสัมผัส NO แทน และปล่อยให้ขา NC ลอย เมื่อมองที่ขา NC กับ COM และ NO กับ COM แล้วจะเห็นว่ามีการทำงานติด-ดับลักษณะคล้ายการทำงานของ สวิชต์ เราสามารถอาศัยคุณสมบัตินี้ไปประยุกต์ใช้งานได้

2.5.2 บอร์ด Relay 8 ช่อง 5 โวลต์



รูปที่ 2.14 บอร์ด Relay 8 ช่อง 5 โวลต์ [5]

จากรูปที่ 2.14 เป็นบอร์ด Relay 8 ช่อง 5 โวลต์บอร์ดรีเลย์ขนาด 8 ช่อง มีเอาต์พุตคอนเน็คเตอร์ที่รีเลย์เป็น NO/COM/NC สามารถใช้กับโหลดได้ทั้งแรงดันไฟฟ้า DC และ AC โดยใช้สัญญาณในการควบคุมการทำงานด้วยสัญญาณโลจิก TTL (5V)

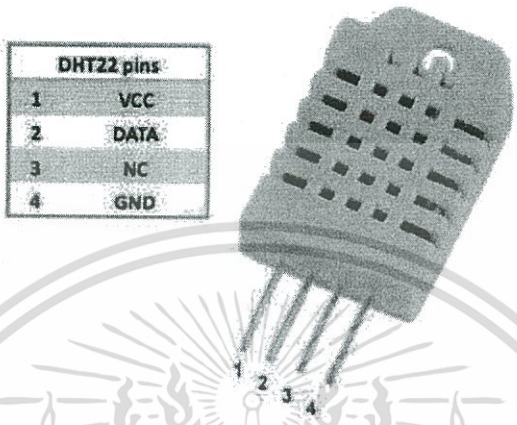
2.5.3 คุณสมบัติบอร์ด Relay 8 ช่อง 5 โวลต์

เป็นรีเลย์เอาต์พุตแบบ SPDT จำนวน 8 ช่องที่สามารถควบคุมไฟ DC ได้สูงสุด 24VDC/7A และควบคุมไฟ AC ได้สูงสุด 220VAC/7A และ 110VAC/10A โดยที่ระดับสัญญาณอินพุตควบคุมแบบ TTL ทำงานด้วยสัญญาณแบบ Active High มี OPTO-ISOLATED เพื่อแยกกราวด์ส่วนของสัญญาณควบคุมกับไฟที่ขับรีเลย์ออกจากกัน มีจัมพ์เปอร์สำหรับเลือกว่าจะใช้กราวด์ร่วมหรือแยก มี LED แสดงสถานะ การทำงานของรีเลย์และแสดงสถานะของบอร์ด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6 โมดูลวัดอุณหภูมิและความชื้น

2.6.1 โมดูลวัดอุณหภูมิและความชื้น DHT



รูปที่ 2.15 โมดูล DHT 22 [6]

ประโยชน์ของSensor
ใช้วัดความชื้นสัมพัทธ์
ใช้วัดอุณหภูมิ

ความชื้น(%) คือปริมาณไอน้ำที่มีอยู่ในอากาศบริเวณใดบริเวณหนึ่ง ซึ่งมีสัดส่วนที่แตกต่างกันไปในแต่ละท้องที่ ถ้าอากาศมีความชื้นต่ำ น้ำก็จะระเหยได้มาก แต่ถ้าอากาศมีความชื้นสูง น้ำก็จะระเหยได้น้อย โดยความชื้นนั้นมีหลายประเภท ได้แก่ ความชื้นสัมบูรณ์, ความชื้นจำเพาะ และความชื้นสัมพัทธ์ แต่เนื่องจากDHT 22 Relative Humidity and Temperature Sensorเป็นsensorที่สามารถวัดได้แค่ ความชื้นสัมพัทธ์ ดังนั้นความชื้นที่เราควรจะรู้จักต่อไปคือ “ความชื้นสัมพัทธ์”

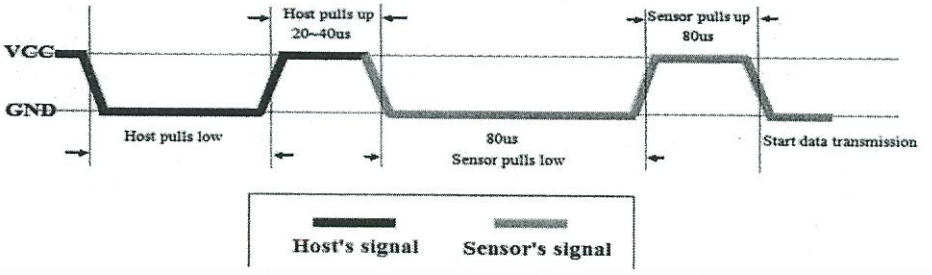
ความชื้นสัมพัทธ์ คืออัตราส่วนของปริมาณไอน้ำในอากาศต่อปริมาณไอน้ำที่ทำให้อากาศอิ่มตัว(อากาศอิ่มตัว คืออากาศที่มีไอน้ำอยู่เต็มที่และไม่สามารถรับเพิ่มได้อีกแล้ว ณ อุณหภูมิหนึ่ง)

ความชื้นสัมพัทธ์ = (ปริมาณไอน้ำที่อยู่ในอากาศ/ปริมาณไอน้ำที่ทำให้อากาศอิ่มตัว) x 100% [6]

อุณหภูมิ คือปริมาณความร้อน ถ้าอากาศร้อน ปริมาณความร้อนก็จะมาก แต่ถ้าอากาศเย็น ปริมาณความร้อนก็จะน้อย

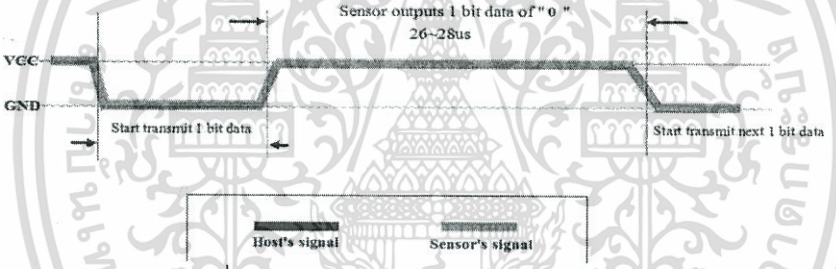
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลักการทำงาน

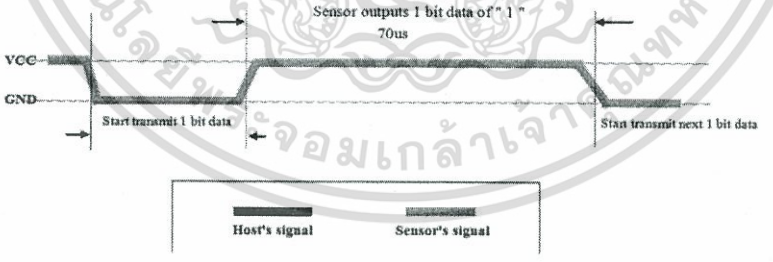


รูปที่ 2.17 สัญญาณของ MCU กับ DHT [6]

จากรูป 2.17 เริ่มจาก MCU จะส่งสัญญาณ pull down voltage ไปยัง DHT22 โดย DHT22 จะใช้เวลาอย่างต่ำ 1 ms และ MCU จะ pull up voltage เพื่อรอการตอบสนองจาก DHT ประมาณ 20-40 us หลังจากนั้น DHT จะส่งสัญญาณ pull down voltage เวลา 80 us เป็นการตอบสนองไปยัง MCU แล้ว DHT ก็จะ pull up voltage เพื่อเตรียมส่งข้อมูล โดยในการส่งข้อมูลแต่ละบิต DHT จะมีการ pull down voltage 50 us



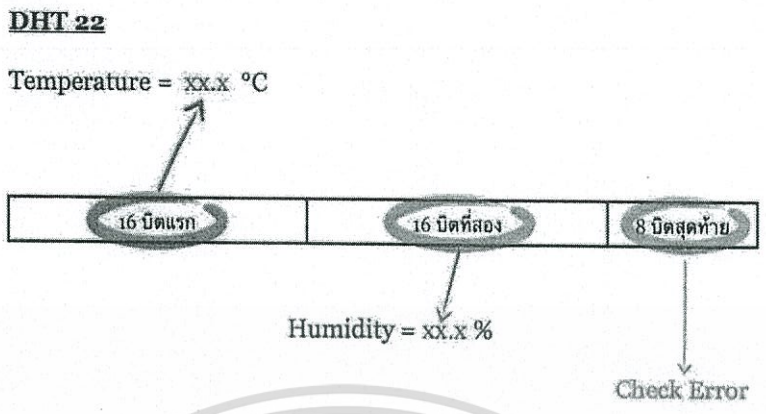
รูปที่ 2.18 การส่งบิตค่า “0” ของ DHT [6]



รูปที่ 2.19 การส่งบิตค่า “1” ของ DHT [6]

จากรูป 2.18 หลังจาก DHT มีการ pull down voltage 50 us เพื่อเป็นการบอก MCU ว่าจะส่งข้อมูล 1 บิต โดยการส่งบิตค่า “0” DHT จะทำการส่งสัญญาณ pull up voltage 26-28 us และ จากรูป 2.19 ส่งบิตค่า “1” DHT จะทำการส่งสัญญาณ pull up voltage 70 us

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.20 บิตข้อมูลของ DHT 22 [6]

การส่งข้อมูลของ DHT22 คือ จะส่งทั้งหมด 40 บิต โดยจะแบ่งเป็น 3 ส่วน สองส่วนแรกส่วนละ 16 บิต และส่วนสุดท้าย 8 บิต ซึ่ง 16บิตแรกและ 16บิตที่สอง หมายถึงค่าอุณหภูมิและค่าความชื้นตามลำดับ ที่รวมทั้งค่าหน้าและหลังทศนิยม โดย ตัวเลขหลักหน่วยจะหมายถึงตัวหลังทศนิยม และ 8บิตสุดท้ายคือเป็นค่าที่ตรวจสอบว่าข้อมูล error หรือไม่

เปรียบเทียบความแตกต่างระหว่าง DHT11 และ DHT22

ความแตกต่าง	DHT11	DHT22
ขอบเขตและความเคลื่อนในการวัดความชื้นสัมพัทธ์	ความชื้น : 20 - 80 % ความคลาดเคลื่อน : 5 %	ความชื้น : 0 - 100 % ความคลาดเคลื่อน : 5 %
ขอบเขตและความเคลื่อนในการวัดอุณหภูมิ	อุณหภูมิ : 0 - 50 °C ความคลานเคลื่อน : ± 2 °C	อุณหภูมิ : -40 - 125 °C ความคลานเคลื่อน : ± 0.5 °C
ความว่องไวในการประมวลผล	1 Hz (1 ครั้งต่อวินาที)	0.5 Hz (2 ครั้งต่อวินาที)
ขนาด	กว้าง x ยาว x สูง : 5.5 mm x 12 mm x 15.5 mm	กว้าง x ยาว x สูง : 7.7 mm x 15.1 mm x 25 mm
ราคา	ประมาณ 100 บาท	ประมาณ 200 บาท

รูปที่ 2.21 เปรียบเทียบความแตกต่างระหว่าง DHT11 และ DHT22 [6]

2.7 RC Servo Motor

Servo เป็นคำศัพท์ที่ใช้กันทั่วไปในระบบควบคุมอัตโนมัติ มาจากภาษาละตินคำว่า Sevus หมายถึง “ทาส” (Slave) ในเชิงความหมายของ Servo Motor ก็คือ Motor ที่เราสามารถ

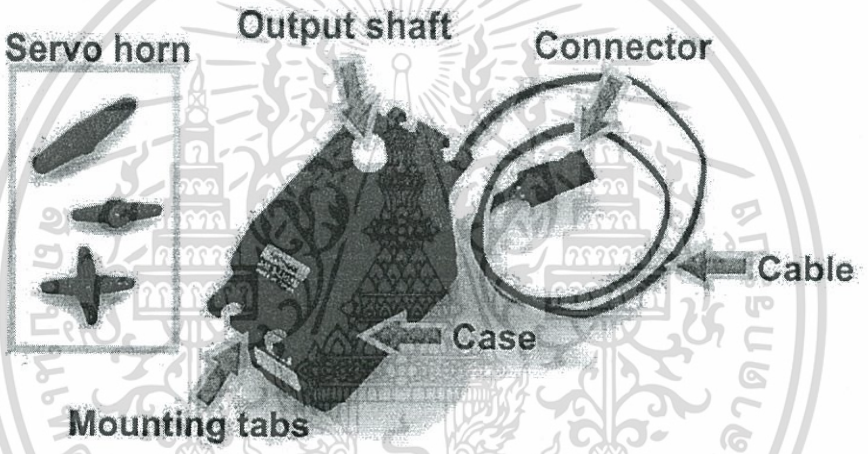
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สั่งงานหรือตั้งค่า แล้วตัว Motor จะหมุนไปยังตำแหน่งองศาที่เราสั่งได้เองอย่างถูกต้อง โดยใช้การควบคุมแบบป้อนกลับ (Feedback Control) ในบทความนี้จะกล่าวถึง RC Servo Motor ซึ่งนิยมนำมาใช้ในเครื่องเล่นที่บังคับด้วยคลื่นวิทยุ (RC = Radio - Controlled) เช่น เรือบังคับวิทยุ รถบังคับวิทยุ เฮลิคอปเตอร์บังคับวิทยุ เป็นต้น

Feedback Control คือ ระบบควบคุมที่มีการวัดค่าเอาต์พุตของระบบนำมาเปรียบเทียบกับค่าอินพุตเพื่อควบคุมและปรับแต่งให้ค่าเอาต์พุตของระบบให้มีค่า เท่ากับ หรือใกล้เคียงกับค่าอินพุต

2.7.1 ส่วนประกอบ RC Servo Motor

2.7.1.1 ส่วนประกอบภายนอก RC Servo Motor



รูปที่ 2.22 ส่วนประกอบภายนอก RC Servo Motor [7]

Case ตัวถัง หรือ กรอบของตัว Servo Motor

Mounting Tab ส่วนจับยึดตัว Servo กับชิ้นงาน

Output Shaft เพลาส่งกำลัง

Servo Horns ส่วนเชื่อมต่อกับ Output shaft เพื่อสร้างกลไกล

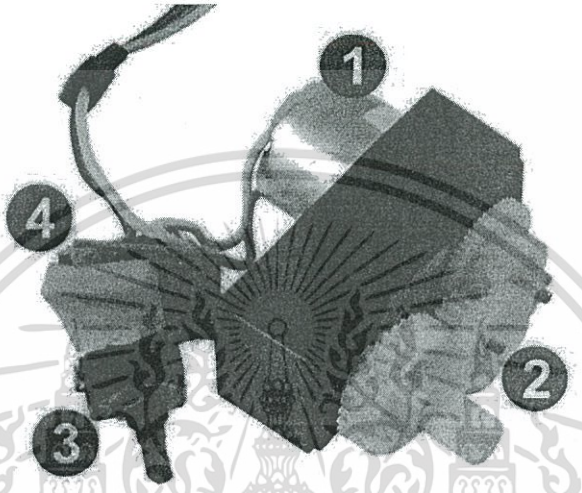
Cable สายเชื่อมต่อเพื่อ จ่ายไฟฟ้า และ ควบคุม Servo Motor จะประกอบด้วยสายไฟ 3 เส้นและใน RC Servo Motor จะมีสีของสายแตกต่างกันไปดังนี้

- สายสีแดง คือ ไฟเลี้ยง (4.8-6V)
- สายสีดำ หรือ น้ำตาล คือ กราวด์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

o สายสีเหลือง (ส้ม ขาว หรือฟ้า) คือ สายส่งสัญญาณพัลซ์ควบคุม (3-5V) Connector จุดเชื่อมต่อสายไฟ

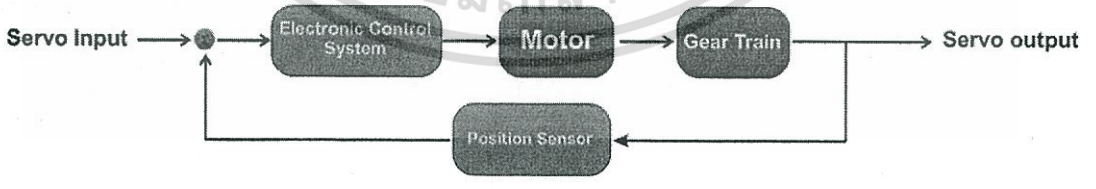
2.7.1.2 ส่วนประกอบภายใน RC Servo Motor



รูปที่ 2.23 ส่วนประกอบภายใน RC Servo Motor [7]

1. Motor เป็นส่วนของตัวมอเตอร์
2. Gear Train หรือ Gearbox เป็นชุดเกียร์ทดแรง
3. Position Sensor เป็นเซ็นเซอร์ตรวจจับตำแหน่งเพื่อหาค่าองศาในการหมุน
4. Electronic Control System เป็นส่วนที่ควบคุมและประมวลผล

Servo Motor Block Diagram



รูปที่ 2.24 Servo Motor Block Diagram [7]

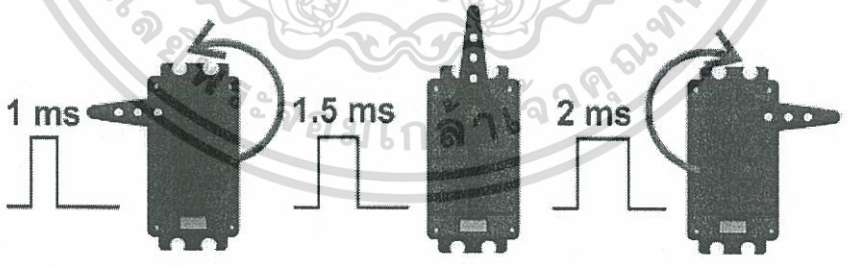
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7.2 หลักการทำงานของ RC Servo Motor

เมื่อจ่ายสัญญาณพัลซ์เข้ามายัง RC Servo Motor ส่วนวงจรควบคุม (Electronic Control System) ภายใน Servo จะทำการอ่านและประมวลผลค่าความกว้างของสัญญาณพัลซ์ที่ส่งเข้ามาเพื่อแปลค่าเป็นตำแหน่งองศาที่ต้องการให้ Motor หมุนเคลื่อนที่ไปยังตำแหน่งนั้น แล้วส่งคำสั่งไปทำการควบคุมให้ Motor หมุนไปยังตำแหน่งที่ต้องการ โดยมี Position Sensor เป็นตัวเซ็นเซอร์คอยวัดค่ามุมที่ Motor กำลังหมุน เป็น Feedback กลับมาให้วงจรควบคุมเปรียบเทียบกับค่าอินพุตเพื่อควบคุมให้ได้ตำแหน่งที่ต้องการอย่างถูกต้องแม่นยำ

2.7.2.1 สัญญาณ RC ในรูปแบบ PWM

ตัว RC Servo Motor ออกแบบมาใช้สำหรับรับคำสั่งจาก Remote Control ที่ใช้ควบคุมของเล่นด้วยสัญญาณวิทยุต่างๆ เช่น เครื่องบินบังคับ รถบังบังคับ เรือบังคับ เป็นต้น ซึ่ง Remote จำพวกนี้ที่ภาครับจะแปลงความถี่วิทยุออกมาในรูปแบบสัญญาณ PWM (Pulse Width Modulation) มุมหรือองศาจะขึ้นอยู่กับความกว้างของสัญญาณพัลซ์ ซึ่งโดยส่วนมากความกว้างของพัลซ์ที่ใช้ใน RC Servo Motor จะอยู่ในช่วง 1-2 ms หรือ 0.5-2.5 ms ยกตัวอย่างเช่นหากกำหนดความกว้างของสัญญาณพัลซ์ไว้ที่ 1 ms ตัว Servo Motor จะหมุนไปทางด้ายซ้ายจนสุด ในทางกลับกันหากกำหนดความกว้างของสัญญาณพัลซ์ไว้ที่ 2 ms ตัว Servo Motor จะหมุนไปยังตำแหน่งขวาสุด แต่หากกำหนดความกว้างของสัญญาณพัลซ์ไว้ที่ 1.5 ms ตัว Servo Motor ก็จะหมุนมาอยู่ที่ตำแหน่งตรงกลางพอดี



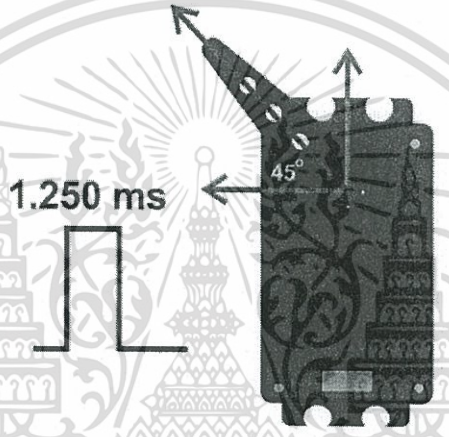
รูปที่ 2.25 มุมหรือองศากับความกว้างของสัญญาณพัลซ์ [7]

ดังนั้นสามารถกำหนดองศาการหมุนของ RC Servo Motor ได้โดยการเทียบค่า เช่น RC Servo Motor สามารถหมุนได้ 180 องศา โดยที่ 0 องศาใช้ความกว้างพัลซ์เท่ากับ 1000 us ที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

180 องศาความกว้างพัลซ์เท่ากับ 2000 us เพราะฉะนั้นค่าที่เปลี่ยนไป 1 องศาจะใช้ความกว้างพัลซ์ต่างกัน (2000-1000)/180 เท่ากับ 5.55 us

จากการหาค่าความกว้างพัลซ์ที่มุม 1 องศาข้างต้น หากต้องกำหนดให้ RC Servo Motor หมุนไปที่มุม 45 องศาจะหาค่าพัลซ์ที่ต้องการได้จาก 5.55 x 45 เท่ากับ 249.75 us แต่ที่มุม 0 องศาเราเริ่มที่ความกว้างพัลซ์ 1ms หรือ 1000 us เพราะฉะนั้นความกว้างพัลซ์ที่ใช้กำหนดให้ RC Servo Motor หมุนไปที่ 45 องศา คือ 1000 + 249.75 เท่ากับประมาณ 1250 us



รูปที่ 2.26 ความกว้างพัลซ์ที่มุม 45 องศา [7]

2.8 Firebase

Firebase คือ Project ที่ถูกออกแบบมาให้เป็น API และ Cloud Storage สำหรับพัฒนา Realtime Application รองรับหลาย Platform ทั้ง IOS App, Android App, Web App **Firebase** ถูกสร้างขึ้นจากคุณสมบัติเสริมว่านักพัฒนาสามารถผสมและจับคู่เพื่อให้พอดี กับความต้องการของตน บริษัท ก่อตั้งขึ้นในปี 2011 โดยแอนดรูลีและเจมส์ เทมปลิน สินค้าเริ่มต้น **Firebase** ป็นฐานข้อมูลเรียลไทม์ซึ่งมี API ที่ช่วยให้นักพัฒนาในการจัดเก็บและซิงค์ข้อมูล โดย Google Firebase 2.0 กูลเกิลได้ซื้อกิจการ **Firebase**และมีการพัฒนาให้สามารถ จากบริการ backend เก็บข้อมูลอย่างเดียว มาเป็น แพลตฟอร์ม ครบวงจรสำหรับนักพัฒนาแอพ รองรับบริการแทบทุกอย่างที่นักพัฒนาแอพต้องใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7.1 Firebase มีบริการอะไรให้บ้าง

1. **Firebase Analytics** บริการวิเคราะห์ข้อมูล ดึงเทคโนโลยีมาจาก Google Analytics แถมยังเปิดให้ใช้ฟรีแบบไม่จำกัดปริมาณข้อมูลใดๆ

2. **Firebase Cloud Messaging (FCM)** ระบบส่งข้อความแจ้งเตือน ใช้งานฟรีไม่จำกัดปริมาณข้อความ

3. **Firebase Storage** บริการพื้นที่เก็บข้อมูล เอาไว้เก็บภาพ วิดีโอ หรือไฟล์ขนาดใหญ่จากแอพของผู้ใช้ สร้างอยู่บน Google Cloud Storage

4. **Firebase Remote Config** ตัวช่วยอัพเดตคอนฟิกของแอพ สำหรับปรับแต่งค่าต่างๆในแอพจากระยะไกล (เช่น เกมที่อยากปรับสมดุลของเกมตลอดเวลา) สามารถใช้ร่วมกับ Firebase Analytics เพื่อกำหนดผู้ใช้งานแยกเป็นกลุ่มๆ ได้

5. **Firebase Crash Reporting** ตัวรายงานการแครชของแอพ รองรับทั้ง iOS และ Android

6. **Firebase Test Lab for Android** บริการทดสอบแอพบนฮาร์ดแวร์จริง

7. **Firebase Notifications** เป็นคอนโซลสำหรับนักพัฒนา เพื่อยิงข้อความผ่าน FCM ไปยังผู้ใช้สำหรับโปรโมทหรือกระตุ้นให้ผู้ใช้กลับมาเปิดแอพของเรา (เช่น แจกของในเกม)

8. **Firebase Dynamic Links** บริการ URL กลางที่สามารถชี้ทางไปยังเพจต่างๆ แปรผันตามอุปกรณ์หรือคุณสมบัติของผู้ใช้ (เช่น แต่ละประเทศกดลิงก์เดียวกัน เข้าคนละเพจกัน)

9. **Firebase Invites** ระบบเชิญเพื่อนมาใช้แอพ มีฟีเจอร์ referral คนชวนได้สิทธิประโยชน์

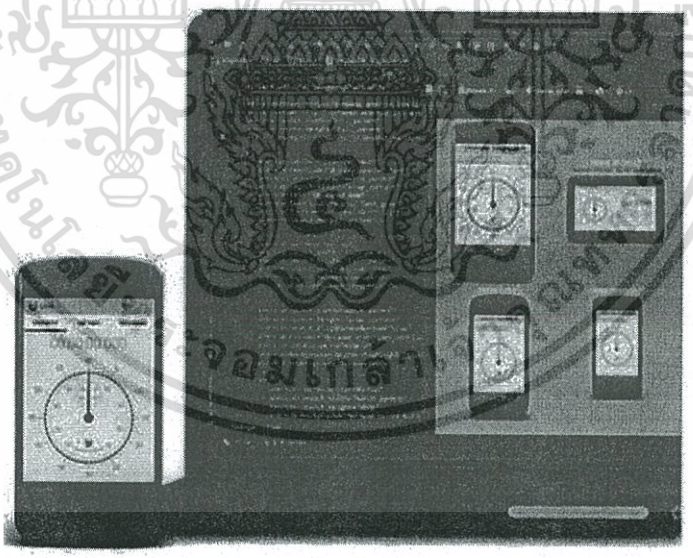
10. **Firebase App Indexing** เปลี่ยนชื่อมาจาก Google App Indexing ที่ช่วยให้ Google Searchค้นเจอเนื้อหาภายในแอพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดังนั้น Firebase จึงครอบคลุมทุกการบริการสำหรับพัฒนา Realtime Application บริการเกือบทุกตัวของ Firebase ใช้งานได้ฟรีแบบไม่จำกัดปริมาณ ยกเว้น Test Lab, Storage, Realtime Database, Hosting ที่คิดเงิน

2.9 Android Studio

รู้จักกับ Android Studio ซึ่งเป็น IDE Tool จาก Google ไว้พัฒนา Android สำหรับ Android Studio เป็น IDE Tools ล่าสุดจาก Google ไว้พัฒนาโปรแกรม Android โดยเฉพาะ โดยพัฒนาจากแนวคิดพื้นฐานมาจาก InteliJ IDEA คล้าย ๆ กับการทำงานของ Eclipse และ Android ADT Plugin โดยวัตถุประสงค์ของ Android Studio คือต้องการพัฒนาเครื่องมือ IDE ที่สามารถพัฒนา App บน Android ให้มีประสิทธิภาพมากขึ้น ทั้งด้านการออกแบบ GUI ที่ช่วยให้สามารถ Preview ตัว App มุมมองที่แตกต่างกันบน Smart Phone แต่ละรุ่น สามารถแสดงผลบางอย่างได้ทันทีโดนไม่ต้องทำการรัน App บน Emulator รวมทั้งยังแก้ไขปรับปรุงในเรื่องของความเร็วของ Emulator ที่ยังเจอปัญหากันอยู่ในปัจจุบัน

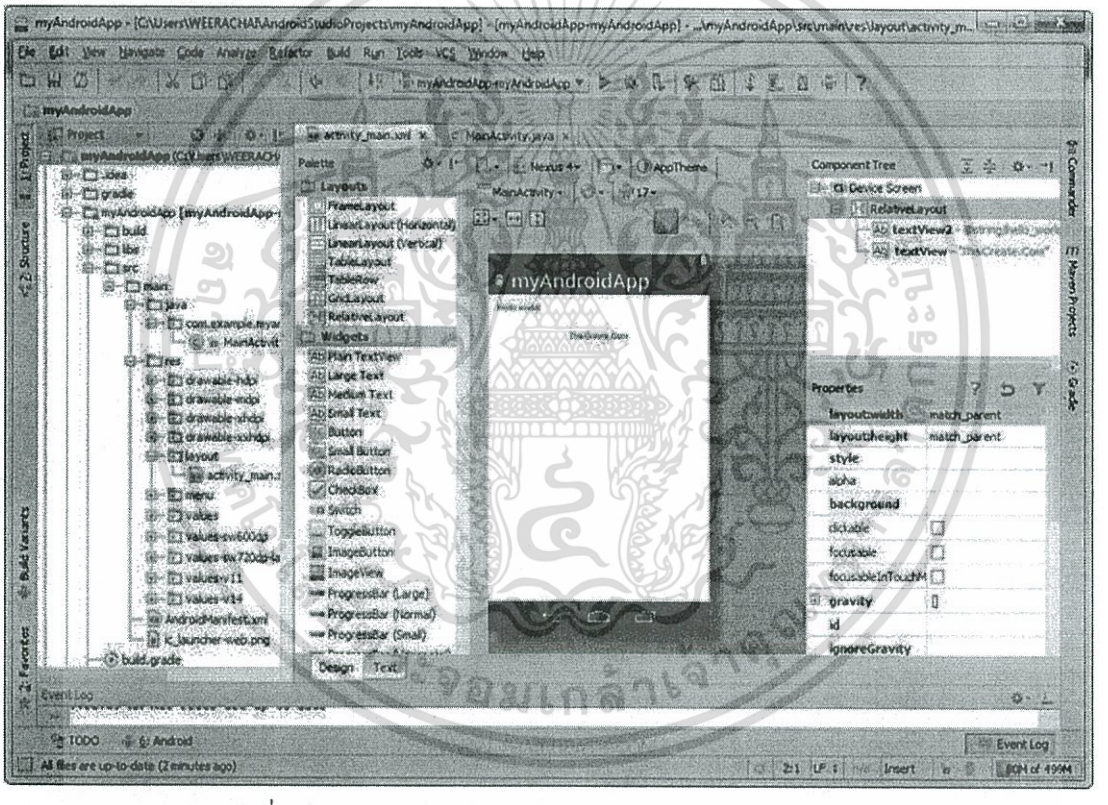


รูปที่ 2.27 Android Studio IDE for Android App [9]

การเขียน Android บน Android Studio จะมีขั้นตอนอยู่ 2 ขั้นตอนก็คือ ติดตั้ง Java SDK และดาวน์โหลด Android Studio มาติดตั้งก็จะสามารถใช้งานได้ทันที โดยที่เรา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไม่ต้องทำการติดตั้ง **Android ADT** Plugin แต่อย่างใด ซึ่งช่วยลดขั้นตอนการติดตั้งเครื่องมือต่าง ๆ ได้ ในปัจจุบัน Android Studio ยังอยู่ในช่วง early access preview แต่เราสามารถดาวน์โหลด **เพื่อใช้งานบน Platform ต่าง ๆ ได้เกือบทุก OS เช่น Windows , Mac และ Linux และจากที่ได้ทำการดาวน์โหลดมาติดตั้งและทดสอบความสามารถของ Android Studio ซึ่งในครั้งแรก ๆ** อาจจะยังงง ๆ ซับสนเมนูต่าง ๆ อยู่บ้าง แต่โดยพื้นฐานแล้วจะคล้าย ๆ กับการเขียน Android บนโปรแกรม Eclipse พวกโครงสร้างไฟล์ หรือ Widgets ต่าง ๆ ก็คล้าย ๆ กัน แต่จะแปลกใหม่ตรงที่มี Preview ในส่วนของ Layout ที่มีความสามารถมากขึ้น เราลองมาดูความสามารถคราว ๆ ของ Android Studio



รูปที่ 2.28 หน้าจอหลักของโปรแกรม Android Studio [9]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.10 ภาษาจาวา (Java)

ภาษาจาวา (Java Language) คือ ภาษาคอมพิวเตอร์ที่ถูกพัฒนาขึ้นโดยบริษัท ซันไมโครซิสเต็มส์(Sun Microsystems) พัฒนามาจากโครงการที่ต้องการพัฒนาระบบซอฟต์แวร์เพื่อควบคุมอุปกรณ์อิเล็กทรอนิกส์ต่างๆ เช่น โทรศัพท์มือถือ โดยมีเป้าหมายการทางานเชื่อมต่อกับอุปกรณ์ฮาร์ดแวร์ต่างๆได้อย่างกว้างขวางและมีประสิทธิภาพ

พื้นฐานการเขียนโปรแกรมภาษาจาวา การเขียนโปรแกรมภาษาจาวา การเขียนโปรแกรมเชิงวัตถุ (Object Oriented Programming : OPP) เป็นการแบ่งขอบเขตของงานออกเป็นส่วนย่อยๆเป็นอิสระต่อกัน แต่มีการทางานร่วมกัน

1) คลาส (Class) คือ การประกาศภาพรวมของกลุ่มวัตถุที่มีคุณลักษณะที่เหมือนกัน มีพฤติกรรมที่เหมือนกัน ในการประกาศคลาส สิ่งที่ต้องทาการประกาศ ประกอบด้วยข้อมูล (Data) และพฤติกรรมที่เหมือนกัน (Behavior) หรือการกระทา (Option)สาหรับคลาสนั้น

2) วัตถุ (Object) คือสิ่งที่เป็นตัวแทนของคลาส มีคุณลักษณะที่บ่งบอกความเป็นตัวเองและสามารถแสดงพฤติกรรมได้

3) เมธอด (Method) เป็นส่วนที่ถูกประกาศไว้ในคลาส ภายในเมธอดจะประกอบด้วย statement ที่เป็นคาสั่งให้โปรแกรมดำเนินการกระทำต่างๆตามที่กำหนดไว้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

การออกแบบและการจัดทำปริญญานิพนธ์

การออกแบบและการจัดทำปริญญานิพนธ์

ในโครงงานนี้จะแบ่งออกได้เป็น 2 ส่วนหลักๆ ได้แก่ ส่วนที่เป็น ไมโครคอนโทรลเลอร์ และ แอปพลิเคชัน ในการทำงานนั้นจะอาศัย Firebase เป็นตัวกลางในการรับและส่งคำสั่งผ่านสองส่วนนี้ให้สามารถทำงานร่วมกันได้โดยผ่านอินเทอร์เน็ต

3.1 การออกแบบ

3.1.1 การควบคุม เปิดและปิดของไมโครคอนโทรลเลอร์

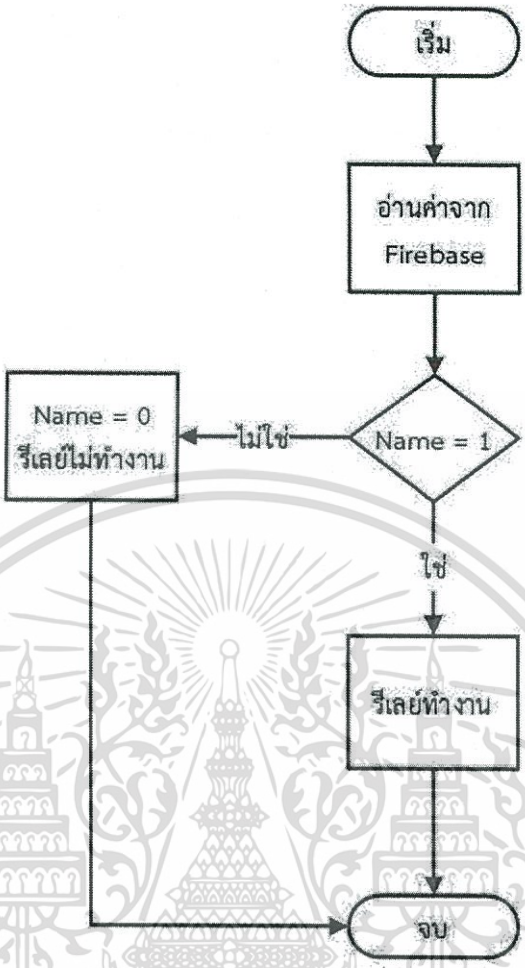
3.1.1.1 การควบคุมผ่านโมดูลรีเลย์

จากการศึกษาการทำงานของโมดูลรีเลย์ เราได้นำมาใช้ในการควบคุมการเปิดและปิดอุปกรณ์รวม 6 ตัวดังนี้

1. พัดลม 2 ตัว
2. หลอดไฟ 1 ดวง
3. ปั๊มน้ำ 2 ตัว

โดยทั้ง 5 ตัวต่อกับโมดูลรีเลย์ในลักษณะเดียวกัน คือ ต่อขั้วบวกเข้ากับขาคอมมอนของรีเลย์ ส่วนขั้วลบต่อเข้า GND และ ขา NO ของรีเลย์ ต่อเข้ากับไฟบวกที่ใช้ในการจ่ายแรงดันให้กับอุปกรณ์ที่ต้องการควบคุม ในการสั่งการนั้นจะให้ Node MCU อ่านค่าจาก Firebase โดยจะตั้งชื่อที่ต่างกันตามอุปกรณ์ที่ต้องการควบคุมเมื่ออ่านค่าได้ จะนำไปสั่งให้ขา pin ที่ต่อกับโมดูลรีเลย์ HIGH หรือ LOW ตามค่าที่อ่านได้และทำให้รีเลย์ทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



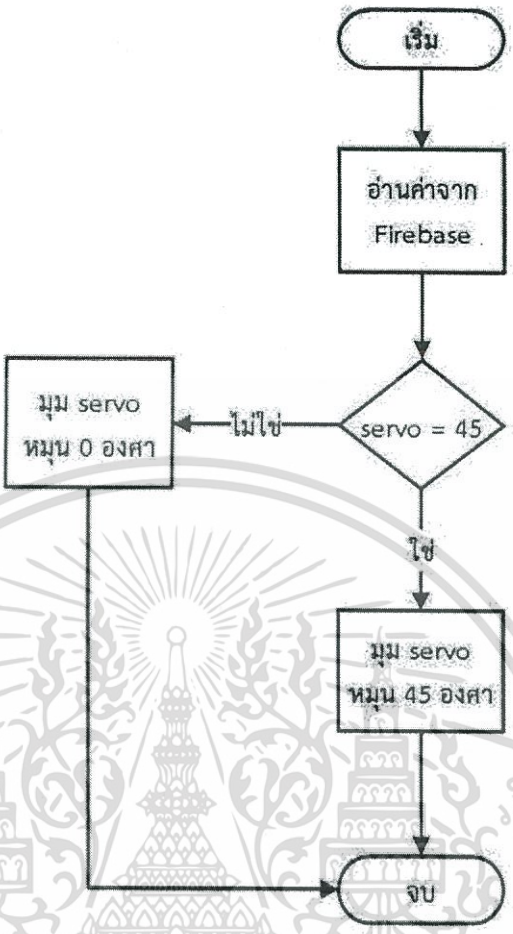
รูปที่ 3.1 แผนผังการสั่งงานรีเลย์ ด้วย Firebase

จากรูปที่ 3.1 เป็นการสั่งการรีเลย์ ด้วยการให้ Node MCU อ่านค่าบน Firebase โดยที่ Name คือ ชื่อข้อมูลบน Firebase ตามอุปกรณ์ เช่น Fan คือ พัดลมตัวที่ 1 ถ้าค่าที่อ่านได้ เป็น 0 Node MCU จะสั่งให้ รีเลย์ทำงานแล้วพัดลมจะเปิด แต่ถ้าค่าที่อ่านได้ เป็น 0 รีเลย์จะไม่ทำงานพัดลมจะปิด เป็นต้น

3.1.1.2 การควบคุม servo motor

จากการศึกษาการทำงานของ servo motor กับ Node MCU นำมาใช้ในการเปิดปิดเครื่องให้อาหาร ใช้วิธีการเดียวกันกับการสั่งงานรีเลย์ผ่าน Firebase

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



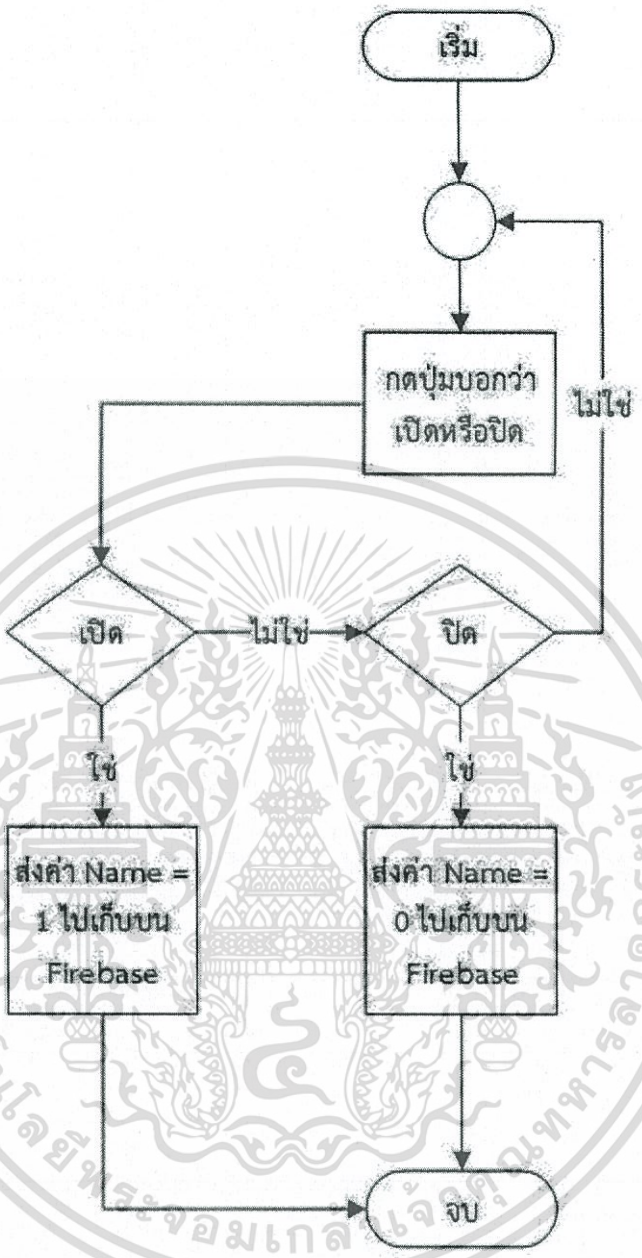
รูปที่ 3.2 แผนผังการสั่งงาน servo motor ด้วย Firebase

จากรูปที่ 3.2 เป็นการสั่งการ servo motor ด้วยการให้ Node MCU อ่านค่าบน Firebase โดยที่ servo คือชื่อข้อมูลบน Firebase ถ้าค่าที่อ่านได้ เป็น 45 Node MCU จะสั่งให้ servo motor หมุนเป็น 45 องศาช่องสำหรับให้อาหารจะเปิด แต่ถ้าค่าที่อ่านได้ เป็น 0 servo motor หมุนเป็น 0 องศาช่องสำหรับให้อาหารจะปิด

3.1.2 การควบคุม เปิดและปิดของแอพพลิเคชัน

ในส่วนการสั่งงานเปิดปิดของแอพพลิเคชันจะเป็นการเขียนค่าลงไปบน Firebase เมื่อมีการกดปุ่มจะไปเซ็ตค่าให้กับข้อมูลบน Firebase ตามอุปกรณ์ที่ต้องการสั่งงาน

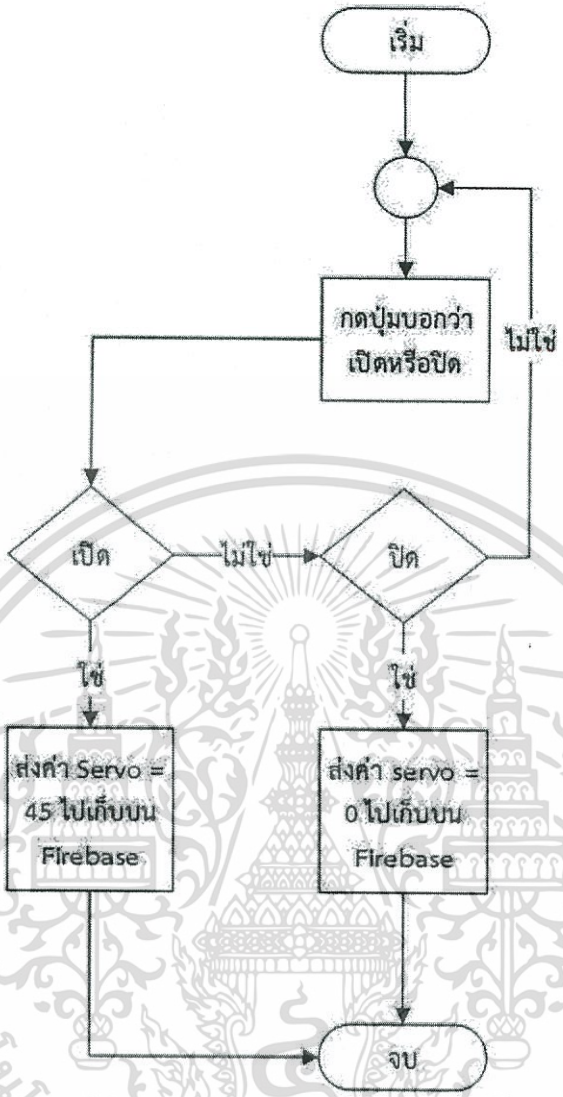
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.3 แผนผังการสั่งงานรีเลย์ ด้วย Firebase ผ่านแอปพลิเคชัน

จากรูปที่ 3.3 เป็นการสั่งการรีเลย์ ด้วย Firebase ผ่านแอปพลิเคชันเมื่อกดปุ่มสั่งงานอุปกรณ์ทั้ง 5 ตัว จะส่งค่าตัวแปรไปเก็บบน Firebase ค่าที่ส่งไปจะไปเก็บบนตัวแปรตามอุปกรณ์ที่ต้องการสั่งงาน เช่น แทน Name ด้วย Fan จะเป็นพัดลมตัวที่ 1 เมื่อกดปุ่มเปิดพัดลมตัวที่ 1 จะส่งค่า Fan เท่ากับ 1 ไปเก็บบน Firebase แต่ถ้า กดปุ่มปิดพัดลมตัวที่ 1 จะส่งค่า Fan เท่ากับ 0 ไปเก็บบน Firebase เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.4 แผนผังการสั่งงาน servo motor ด้วย Firebase ผ่านแอปพลิเคชัน

จากรูปที่ 3.5 เป็นการสั่งการรีเลย์ ด้วย Firebase ผ่านแอปพลิเคชันเมื่อกดปุ่มสั่งงานอุปกรณ์ให้อาหาร จะส่งค่า servo ไปเก็บบน Firebase โดยที่ servo คือชื่อข้อมูลบน Firebase ถ้ากดปุ่มเปิดค่าที่ส่งจะเป็น 45 ค่า servo บน Firebase จะเป็น 45 แต่ถ้ากดปุ่มปิดค่าที่ส่ง เป็น 0 ค่า servo บน Firebase จะเป็น 0

3.1.3 การอ่านค่าอุณหภูมิและความชื้น

3.1.3.1 การอ่านค่าอุณหภูมิและความชื้นจาก DHT22 ด้วย Node MCU

จากการศึกษาการใช้งาน DHT22 ทำการการต่อวงจรและอ่านค่าอุณหภูมิความชื้นแล้วส่งค่าอุณหภูมิและความชื้นไปเก็บบน Firebase

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

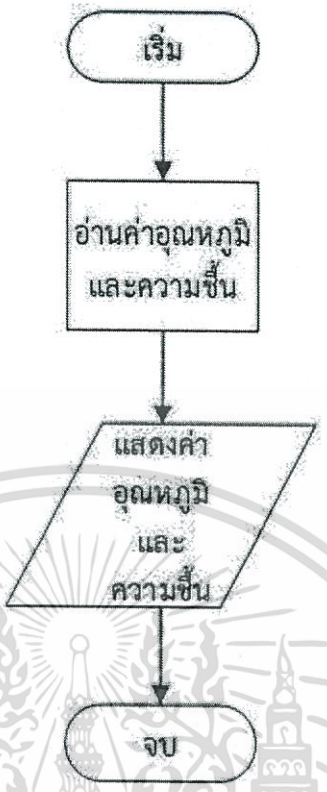
รูปที่ 3.5 แผนผังการส่งค่าอุณหภูมิและความชื้นจาก Node MCU ไปเก็บบน Firebase

จากรูปที่ 3.5 เป็นการส่งค่าอุณหภูมิและความชื้นของ dht22 ที่ Node MCU อ่านได้ไปเก็บบน Firebase โดยจะส่งค่าไปสองค่า คือ อุณหภูมิ เก็บที่ temp และ ความชื้น เก็บที่ humu ของ Firebase

3.1.3.1 การอ่านค่าอุณหภูมิและความชื้นจาก Firebase ด้วยแอปพลิเคชัน

ค่าอุณหภูมิและความชื้นบน Firebase จะถูกอ่านและนำมาแสดงบนหน้าแอปพลิเคชันค่าอุณหภูมิและความชื้นที่อ่านได้นั้นจะเปลี่ยนไปตามค่าบน Firebase

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.6 แผนผังการอ่านค่าอุณหภูมิและความชื้นจาก Firebase มาแสดงที่ แอปพลิเคชัน

จากรูปที่ 3.6 เป็นการอ่านค่าอุณหภูมิและความชื้นของ dht22 ที่ Node MCU อ่านได้และส่งมาเก็บบน Firebase โดยค่าที่อ่านมีสองค่า คือ อุณหภูมิ หรือ temp และ ความชื้น หรือ humu ของ Firebase จากนั้นนำค่าที่ได้มาแสดงบนแอปพลิเคชัน

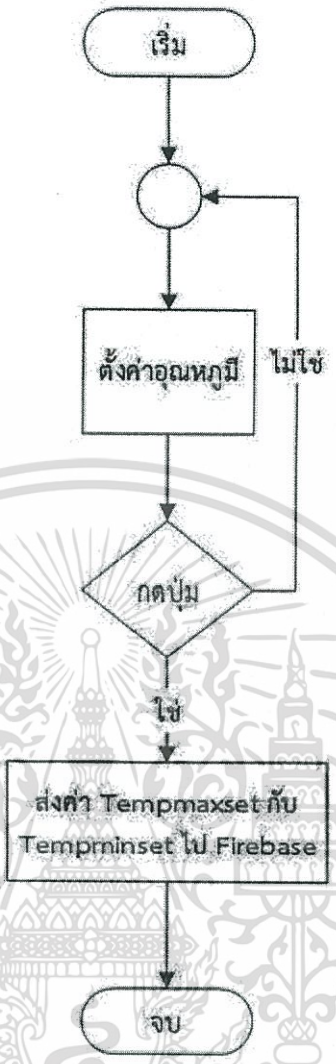
3.1.4 การแจ้งเตือน

ระบบการแจ้งเตือนจะมีอยู่ 2 อย่างคือ แจ้งเตือนค่าอุณหภูมิ และ แจ้งเตือนระดับน้ำ โดยกระบวนการแจ้งเตือนนั้นจะทำงานโดยอาศัยการตัดสินใจของ Node MCU แล้วส่งค่าขึ้น Firebase แล้วจึงให้ แอปพลิเคชันอ่านค่าคำสั่งแจ้งเตือน

3.1.4.1 การตั้งค่าการแจ้งเตือนอุณหภูมิ

จะมีอยู่สามส่วน ส่วนแรก คือ แอปพลิเคชันที่ใช้ตั้งค่า เมื่อเซ็ตค่าอุณหภูมิสูงสุดและอุณภูมิต่ำสุดแล้วกดปุ่มยืนยันค่าอุณหภูมิจะที่เซ็ตไว้จะถูกส่งไปเก็บบน Firebase โดยที่ค่าอุณหภูมิสูงสุดจะเก็บที่ตัวแปร Tempmaxset และ ค่าอุณหภูมิต่ำสุดจะเก็บที่ตัวแปร Tempminset

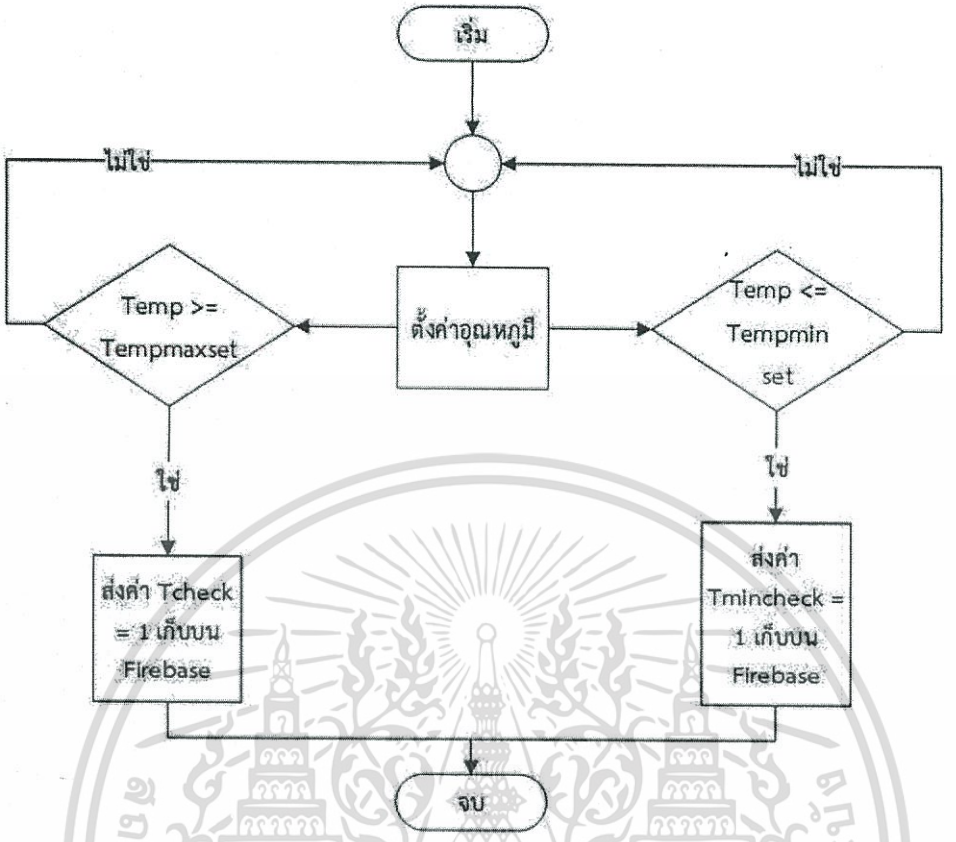
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.7 แผนผังการตั้งค่าการแจ้งเตือนอุณหภูมิบนแอปพลิเคชัน

จากรูปที่ 3.7 เป็นการตั้งค่าอุณหภูมิสูงสุดและต่ำสุด ที่หน้าแอปพลิเคชันจะมี Seekbar สองอันให้ใช้ในการตั้งค่าอุณหภูมิสูงสุดและต่ำสุด เมื่อตั้งค่าเสร็จแล้ว กดปุ่มยืนยัน ค่าจะถูกเซ็ตไปบน Firebase ซึ่งค่าอุณหภูมิสูงสุดจะถูกเก็บไว้ที่ Tempmaxset และ ค่าอุณหภูมิต่ำสุดจะถูกเก็บไว้ที่ Tempminset ส่วนที่สอง คือ หลังจากที่บันทึกค่าอุณหภูมิสูงสุดและต่ำสุดที่ได้จากการตั้งค่าในแอปพลิเคชันลงบน Firebase เป็นที่เรียบร้อยแล้ว Node MCU จะทำการอ่านค่าอุณหภูมิที่เซ็ตจากแอปพลิเคชันซึ่งคือ ค่า Tempmaxset และ Tempminset จากนั้นนำค่าที่ได้มาเปรียบเทียบกับอุณหภูมิที่อ่านได้จาก DHT22 คือ Temp ใน Node MCU

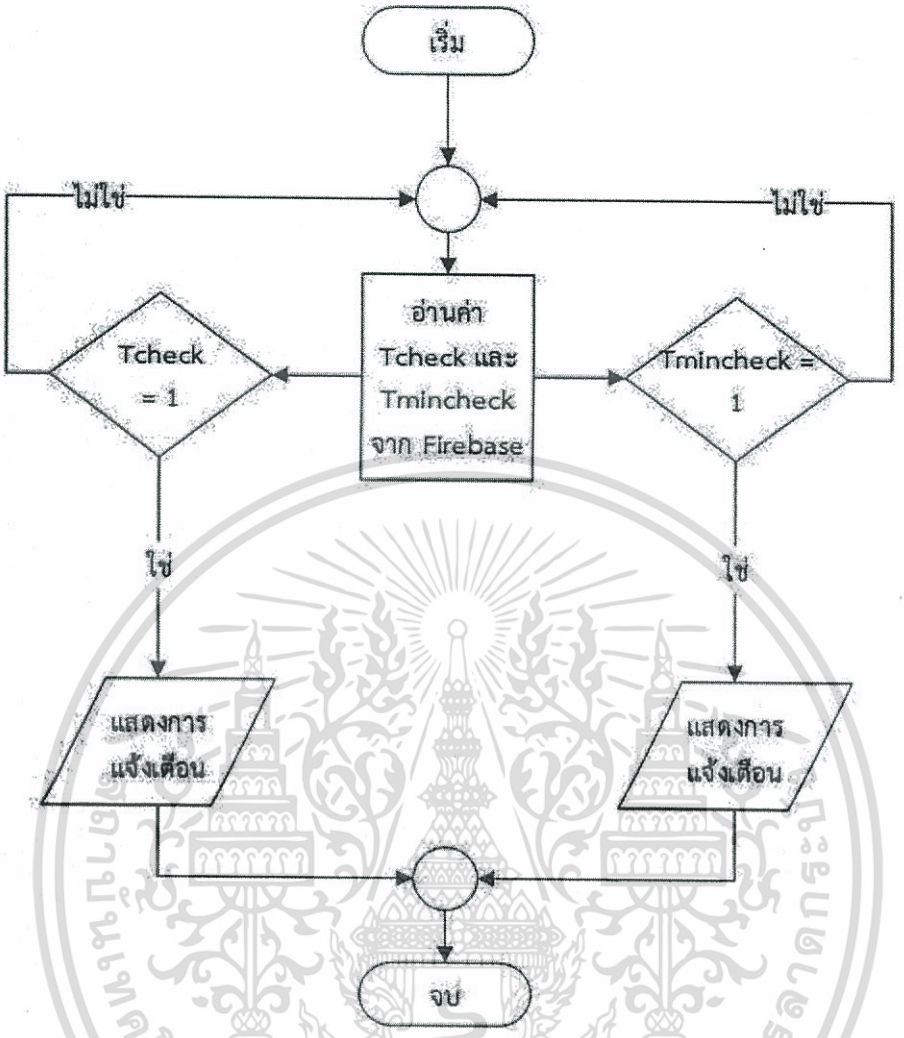
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.8 แผนผังการตั้งค่าการแจ้งเตือนอุณหภูมิบนแอปพลิเคชัน

จากรูปที่ 3.8 เป็นการอ่านค่าอุณหภูมิสูงสุดสุด คือ Tempmaxset และต่ำสุด คือ Tempminset บน Firebase จากนั้นนำมาเปรียบเทียบกับค่าอุณหภูมิที่อ่านได้จาก DHT22 คือ ค่า Temp โดยเมื่อค่า Temp มากกว่าหรือเท่ากับค่า Tempmaxset จะทำการส่งค่า Tcheck = 1 ส่งไปเก็บบน Firebase แต่ถ้าค่า Temp มีค่าน้อยกว่าค่า Tempmaxset จะทำการส่งค่า Tcheck = 0 ส่งไปเก็บบน Firebase และ ในกรณีของ Tempminset เมื่อค่า Temp น้อยกว่าหรือเท่ากับค่า Tempminset จะทำการส่งค่า Tmincheck = 1 ส่งไปเก็บบน Firebase แต่ถ้าค่า Temp มีค่ามากกว่าค่า Tempmin set จะทำการส่งค่า Tcheck = 0 ส่งไปเก็บบน Firebase เพื่อเป็นการกำหนดคำสั่งในการแจ้งเตือนให้อยู่ใน Firebase ส่วนสุดท้าย นั้นเป็นการแจ้งเตือนบนแอปพลิเคชันเมื่อค่าอุณหภูมิถึงกำหนดโดยการอ่านค่า TCheck และ TminCheck จากบน Firebase จะเป็นการอ่านค่าออกมาแล้วเช็คว่าต้องแจ้งเตือนหรือไม่ถ้าเป็นค่าที่ต้องแจ้งเตือนแอปพลิเคชันจะทำการแจ้งเตือนขึ้นมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.9 แผนผังแสดงแจ้งเตือนบนแอปพลิเคชันเมื่อได้รับคำสั่ง

จากรูปที่ 3.9 เป็นการแจ้งเตือนบนแอปพลิเคชันตามเงื่อนไขโดยเมื่ออ่านค่า TCheck เป็น 1 แสดงว่าอุณหภูมิที่อ่านจาก DHT22 มีค่ามากกว่าหรือเท่ากับที่ทำการตั้งค่าอุณหภูมิสูงสุดไว้แล้วจะทำการแจ้งเตือนบนแอปพลิเคชันว่าอุณหภูมินั้นมีค่ามากกว่าหรือเท่ากับที่กำหนด แต่ถ้าค่าที่อ่านจาก TCheck เป็น 0 แสดงว่าอุณหภูมิที่อ่านจาก DHT22 มีค่าน้อยกว่าที่ทำการตั้งค่าอุณหภูมิสูงสุดไว้แล้ว จะไม่มีการแจ้งเตือนบนแอปพลิเคชันและ เมื่ออ่านค่า TminCheck เป็น 1 แสดงว่าอุณหภูมิที่อ่านจาก DHT22 มีค่าน้อยกว่าที่ทำการตั้งค่าอุณหภูมิต่ำสุดไว้แล้วจะทำการแจ้งเตือนบนแอปพลิเคชันว่าอุณหภูมินั้นมีค่าน้อยกว่าหรือเท่ากับที่กำหนด แต่ถ้าค่าที่อ่านจาก TCheck เป็น 0 แสดงว่าอุณหภูมิที่อ่านจาก DHT22 มีค่ามากกว่าที่ทำการตั้งค่าอุณหภูมิต่ำสุดไว้แล้วจะไม่มีการแจ้งเตือนบนแอปพลิเคชัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

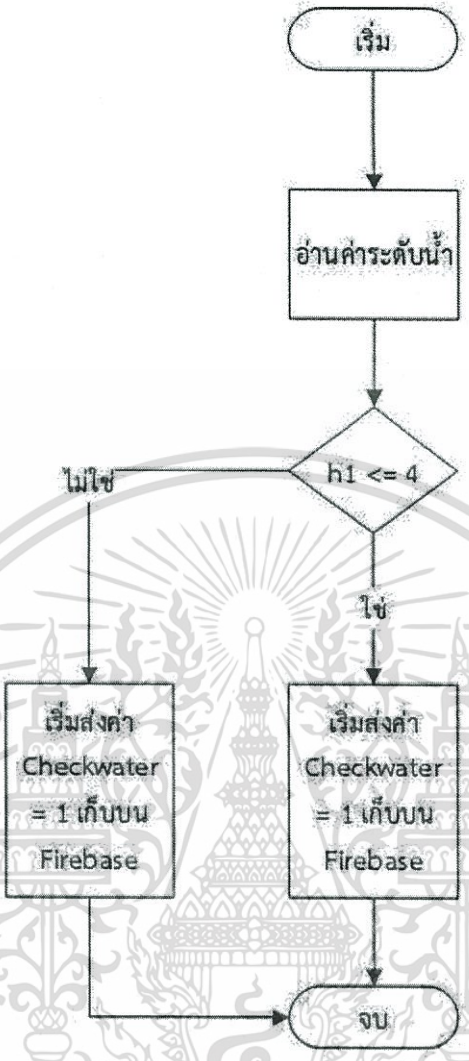
3.1.4.2 การแจ้งเตือนระดับน้ำ

เป็นการแจ้งเตือนระดับน้ำของถังให้น้ำโดยเมื่อระดับน้ำต่ำกว่าเกณฑ์ที่กำหนดไว้จะทำการแจ้งเตือนขึ้นที่แอปพลิเคชันการทำงานจะมีอยู่สามส่วน คือ ส่วนแรกเป็นการอ่านค่าระดับน้ำ ส่วนที่สองเป็นการตัดสินใจว่าถึงเกณฑ์ที่กำหนดหรือไม่ และ ส่วนสุดท้ายเป็นการแจ้งเตือนผ่านแอปพลิเคชัน

รูปที่ 3.10 แผนผังแสดงการอ่านค่าระดับน้ำ

จากรูปที่ 3.10 เป็นการอ่านค่าระยะทางที่ได้จาก HC-RO4 ค่าที่ได้เป็นค่าระยะจากผิวน้ำจนถึงเซ็นเซอร์ดังนั้นต้องนำค่าความสูงของถังให้น้ำมาลบกับค่าที่วัดได้เพื่อให้ได้ค่าความสูงของระดับน้ำ ส่วนที่สองนี้เป็นการนำค่าระดับน้ำที่ได้มาเปรียบเทียบกับเกณฑ์ที่กำหนดไว้ถ้าต่ำกว่าหรือเท่ากับที่กำหนดจะส่งค่าขึ้นไปบน Firebase เพื่อแจ้งเตือน และ ถ้ามีค่ามากกว่าจะส่งค่าขึ้นไปบน Firebase เพื่อไม่ให้แจ้งเตือน

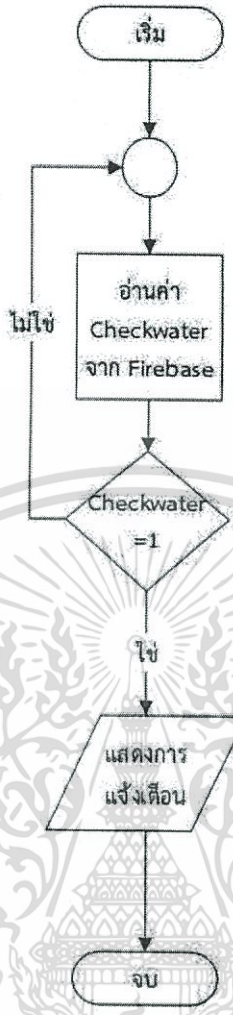
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.11 แผนผังแสดงการอ่านค่าระดับน้ำ

จากรูปที่ 3.11 เป็นการนำค่าระดับน้ำที่อ่านได้มาตัดสินใจว่ามีค่าต่ำกว่าหรือเท่ากับกำหนดหรือไม่ ถ้ามีค่าต่ำกว่าหรือเท่ากับที่กำหนดในที่นี้ใช้ 4 เซนติเมตร Node MCU ESP8266 จะทำการส่งค่า CheckWater เท่ากับ 1 ไปเก็บบน Firebase แต่ถ้ามีค่ามากกว่าที่กำหนดจะทำการส่งค่าCheckWater เท่ากับ 0 ไปเก็บบน Firebase ส่วนสุดท้ายเป็นการแจ้งเตือนของแอปพลิเคชันเมื่อค่าถึงที่กำหนดจะทำการแจ้งเตือนโดยการอ่านค่า CheckWater ของ Firebase แล้วมาตัดสินใจ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.12 แผนผังแสดงการแจ้งเตือนเมื่อระดับถึงค่าที่กำหนด

จากรูปที่ 3.12 เป็นการแจ้งเตือนระดับน้ำโดยการอ่านค่า CheckWater เมื่อมีค่าเป็น 1 จะแจ้งเตือนขึ้นมาว่าค่าถึงที่กำหนด แต่ถ้าค่าที่อ่านได้นั้นเป็น 0 จะไม่มีการแจ้งเตือนเกิดขึ้น

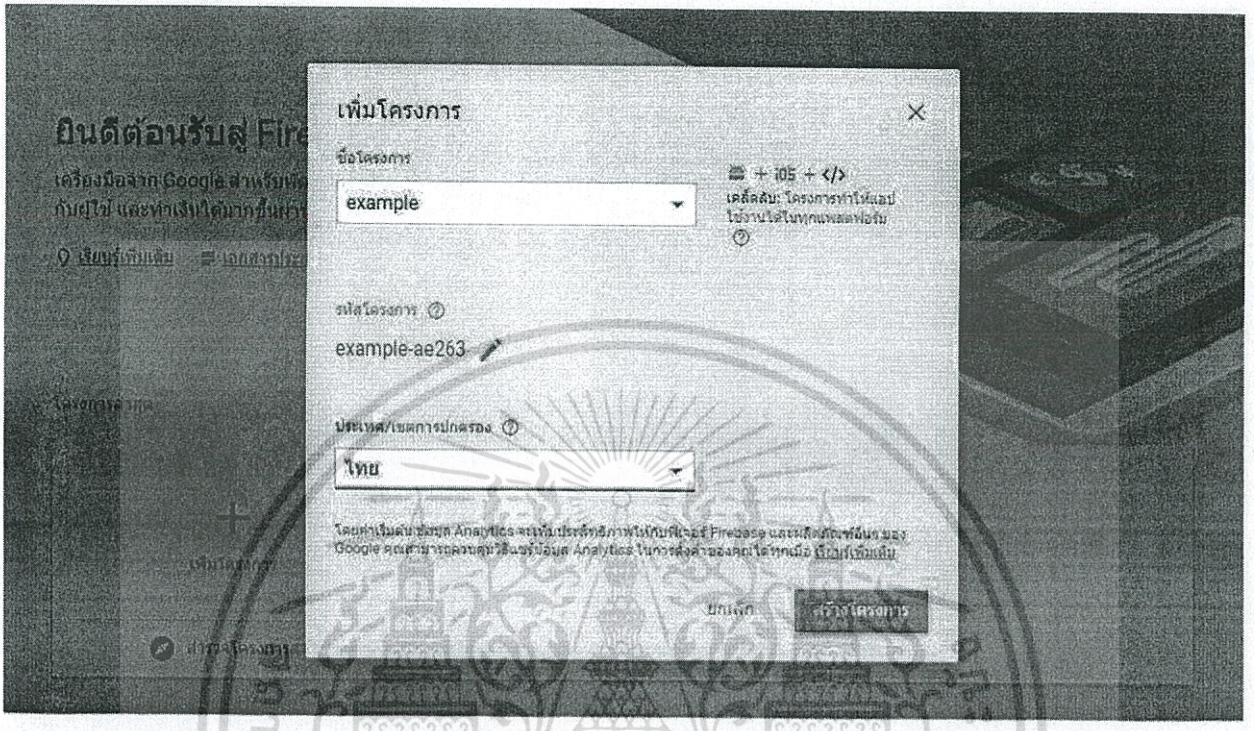
3.1.5 การใช้งาน Firebase

3.1.5.1ขั้นตอนการใช้งาน Firebase Realtime Database

ขั้นแรกต้องทำการ log in เข้า firebase ก่อนโดยสามารถใช้บัญชีของ google ในการเข้าใช้งานได้เลยไม่ยุ่งยาก

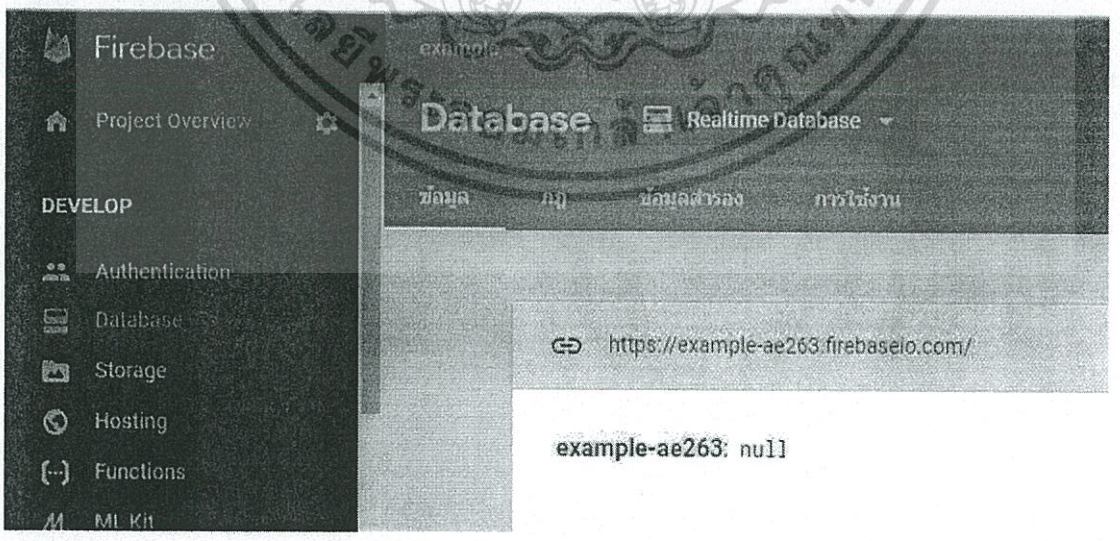
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นที่สองเมื่อเข้ามาที่หน้าแรกให้ทำการกดปุ่มเพิ่มโครงการเพื่อเป็นการสร้างโครงการของเราขึ้นมาให้เราใส่ชื่อโครงการและประเทศของเราจากนั้นกดสร้างโครงการโครงการของเราใช้ชื่อว่า example



รูปที่ 3.13 การเพิ่มโครงการของ Firebase

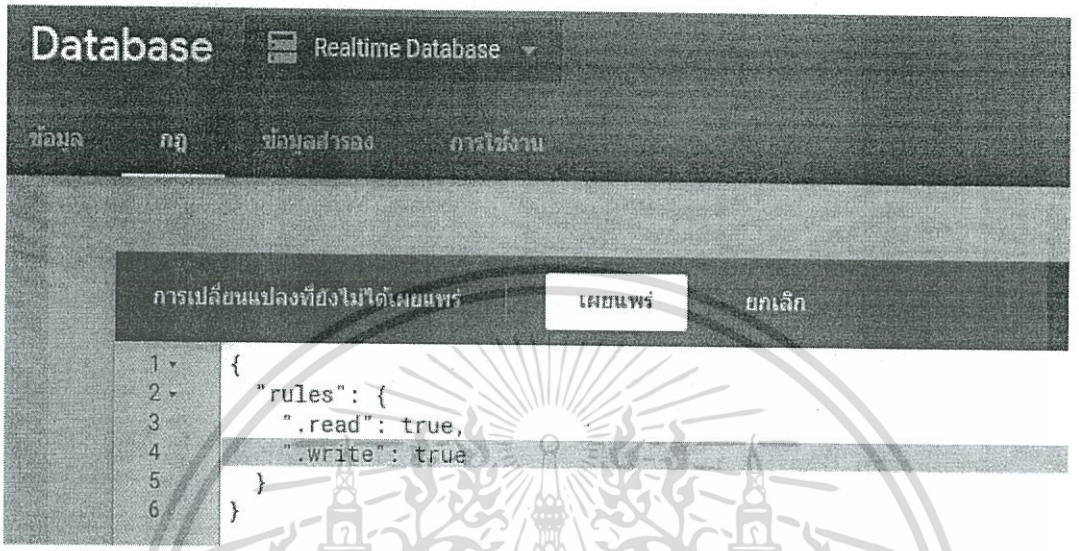
ขั้นที่สามเมื่อสร้างโครงการเรียบร้อยแล้วต่อไปจะเป็นการตั้งค่า Firebase Realtime database ให้กดไปที่เมนู database ด้านซ้ายมือ



รูปที่ 3.14 การใช้งาน Realtime database

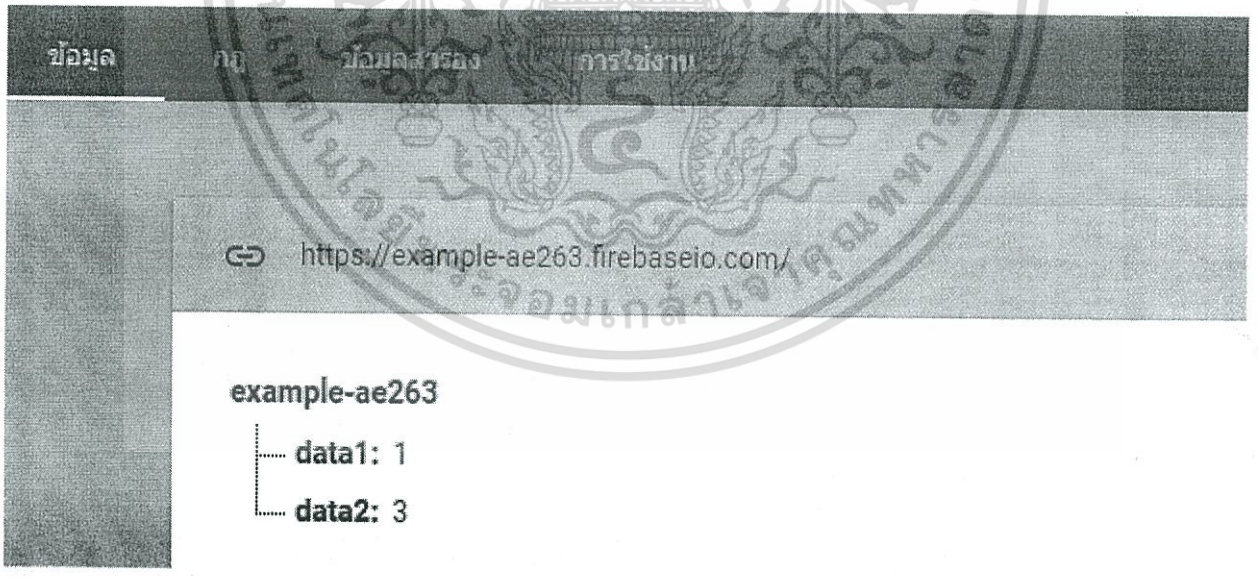
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากนั้นให้ไปตั้งค่าที่ กฎ ก่อนเนื่องจากเรายังไม่สามารถเขียนหรืออ่านค่าบน Realtime database ได้โดยให้เปลี่ยน false เป็น true ทั้ง read และ write แล้วทำการกดเผยแพร่ก็จะสามรถใช้งาน Realtime database ได้



รูปที่ 3.15 การตั้งค่ากฎใน Realtime database

ขั้นสุดท้ายทดลองเขียนค่าลงบน Realtime database เราเขียนข้อมูลใส่เข้าไปสองค่าคือ data1 และ data2

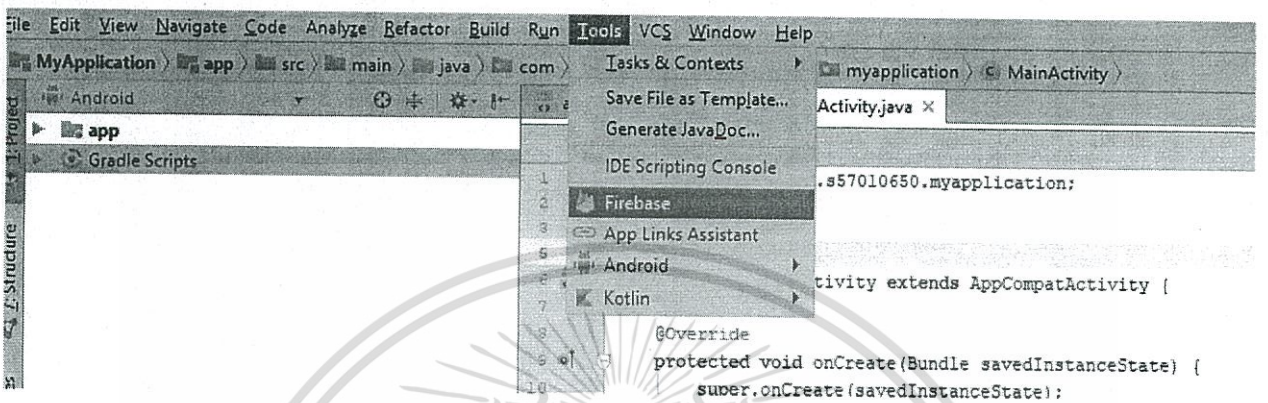


รูปที่ 3.16 การเพิ่มข้อมูลลงบน Realtime database

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

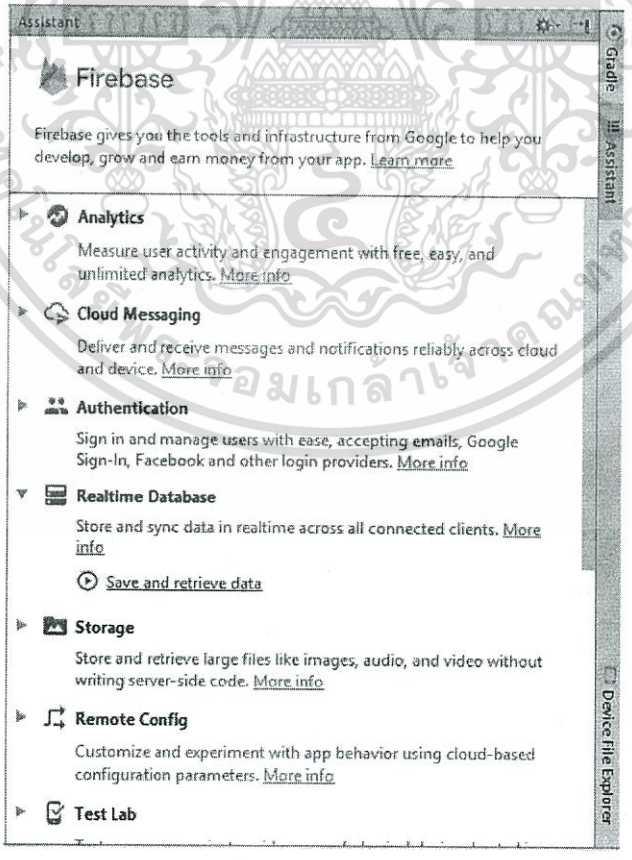
ขั้นตอนการเชื่อมต่อ Firebase Realtime database กับ Android studio

ขั้นแรกเมื่อเราทำการเปิด Android studio ให้เราทำการเปิดโปรเจคที่เราต้องการเชื่อมต่อขึ้นมาจากนั้นให้ไปที่เมนู Tools ที่อยู่ด้านบนแล้วกดตรง Firebase



รูปที่ 3.17 การเชื่อมต่อ Android studio กับ Firebase

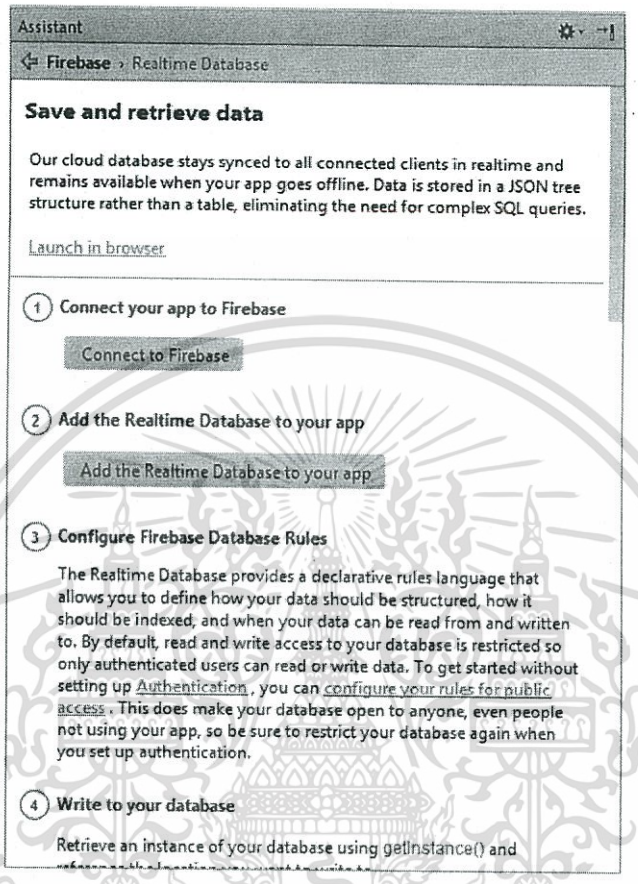
ขั้นที่สองหลังจากเลือก Firebase ที่เมนู Tools แล้วหน้า Assistant ของ Firebase จะปรากฏขึ้นมาด้านขวา ให้ทำการเลือก Realtime Database จะเจอกับคำว่า Save and retrieve data ให้กดเข้าไป



รูปที่ 3.18 หน้า Assistant ของ Firebase

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

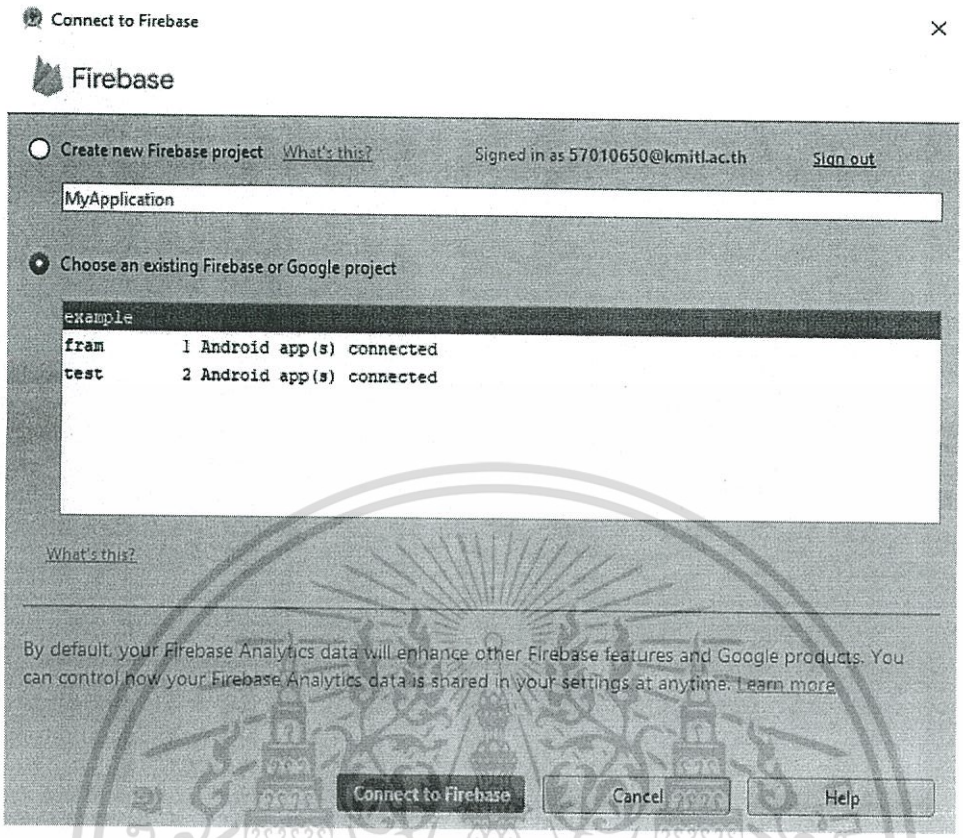
ขั้นที่สามให้ทำการกด Connect to Firebase ที่ Connect your app to Firebase แล้วทำการ log in ด้วยบัญชี google เดียวกับที่ log in ใน Firebase



รูปที่ 3.19 หน้าเชื่อมต่อ Realtime Database บน Android studio

ที่หน้า Connect to Firebase ให้เลือกชื่อ Realtime Database ที่ตั้งไว้ใน Firebase ซึ่งที่เราตั้งไว้คือ example จากนั้นกด Connect to Firebase

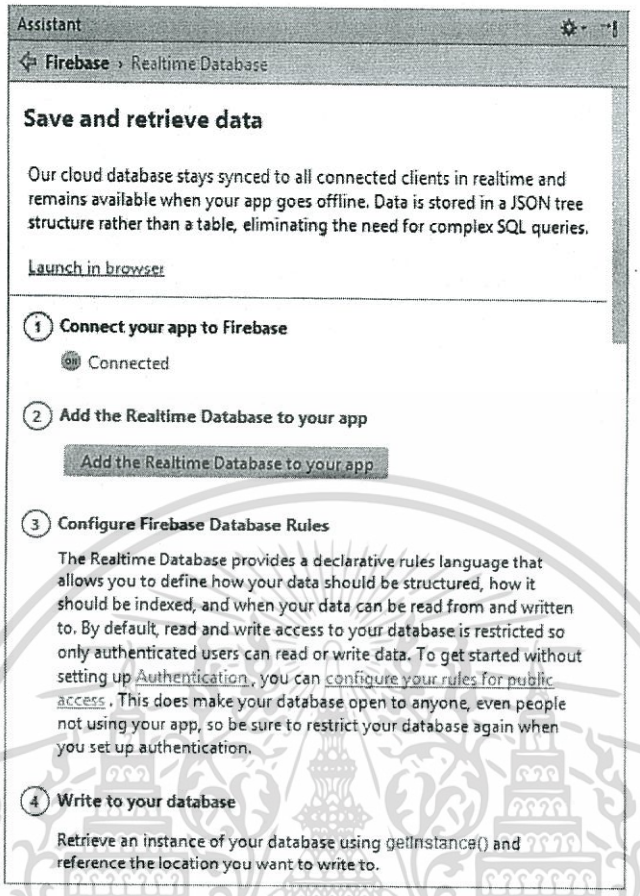
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.20 หน้าต่างสำหรับเลือกการเชื่อมต่อกับ Firebase

ขั้นตอนที่สุดท้ายหลังจากกด Connect to Firebase **ที่** Connect your app to Firebase **จะขึ้นคำว่า** Connected **ซึ่งหมายถึงการเชื่อมต่อกับ** Firebase **เรียบร้อยแล้วจากนั้นที่** Add the Realtime Database to your app **ให้กดเข้าไปที่** Add the Realtime Database to your app

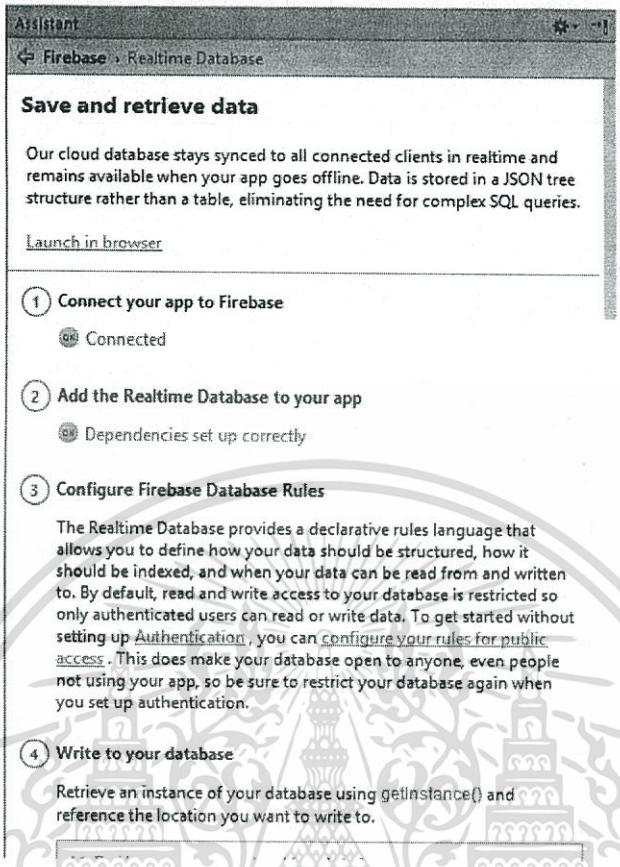
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.21 หน้าต่างเชื่อต่อกับ Firebase เรียบร้อยแล้ว

เมื่อกดแล้วจะทำการเพิ่ม Realtime Database ไปที่ app เมื่อเสร็จสิ้นจะขึ้นคำว่า Dependencies set up correctly เป็นอันเสร็จการเชื่อมต่อ Android studio กับ Firebase Realtime Database

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.22 หน้าเชื่อมต่อ Realtime database หลังจากเพิ่ม Realtime Database เรียบร้อยแล้ว

3.2 เครื่องมือที่ใช้ในการทดลอง

ในการออกแบบโครงงานนั้นจะมีทั้งส่วนที่เป็นฮาร์ดแวร์ และ ซอร์ฟแวร์ โดยแบ่งโปรแกรมซอร์ฟแวร์ และ ฮาร์ดแวร์ ที่ใช้ในการออกแบบได้ดังนี้

ฮาร์ดแวร์ที่ใช้มีดังนี้

- Broad Node MCU ESP8266
- Relay module
- EFDV245 Futaba S3003 Servo Motor
- Dht22
- HC-SR04
- ปั๊มน้ำขนาดเล็ก 220v
- พัดลม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- หลอดไฟ

ซอร์ฟแวร์ที่ใช้มีดังนี้

- Arduino IDE
- Android Studio
- Firebase Realtime database

3.3 การจัดเก็บผลการทดลอง

3.3.1 การทดสอบการวัดค่าอุณหภูมิและความชื้น

3.3.2 การทดสอบการแสดงผลค่าอุณหภูมิบน Smart phone

3.3.3 การทดสอบการเปิดปิดด้วยแอปพลิเคชัน

3.3.4 การทดสอบการวัดค่าระดับน้ำ

3.3.5 การทดสอบการแจ้งเตือนผ่าน Smart phone

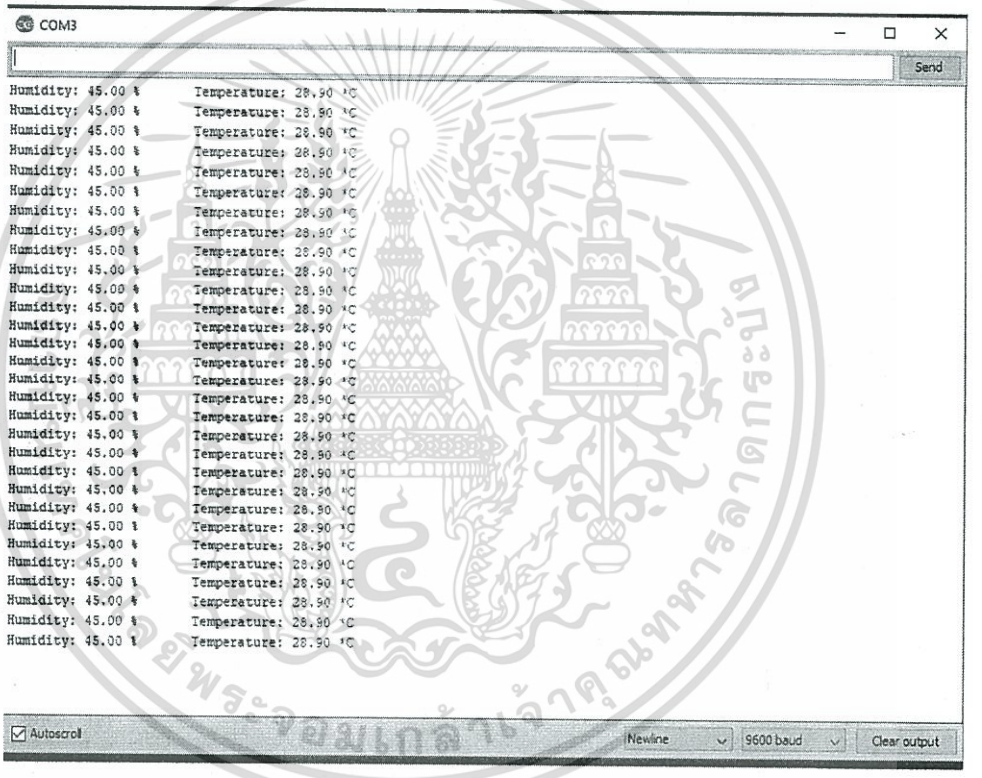
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

ผลการทดลอง

4.1 การวัดค่าอุณหภูมิและความชื้น

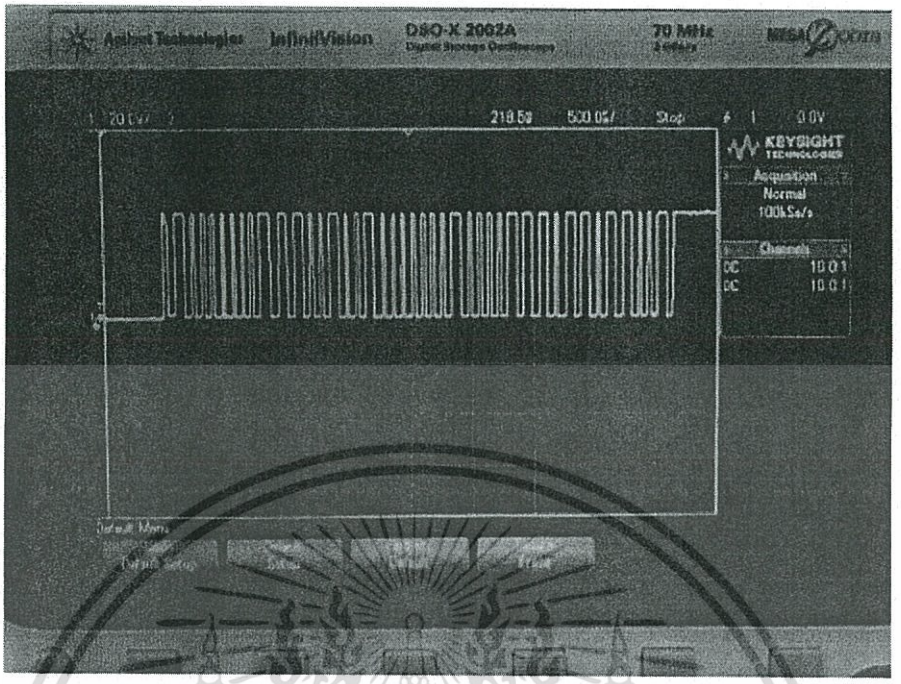
ในการวัดค่าอุณหภูมิและความชื้นนั้นจะใช้ DHT22 ในการวัดโดยใช้ Node MCU ESP8266 เป็นตัวนำค่ามาแสดงจากนั้นนำค่าที่อ่านได้ส่งขึ้นไป Realtime database ของ Firebase



รูปที่ 4.1 ค่าอุณหภูมิและความชื้นบน Serial Monitor

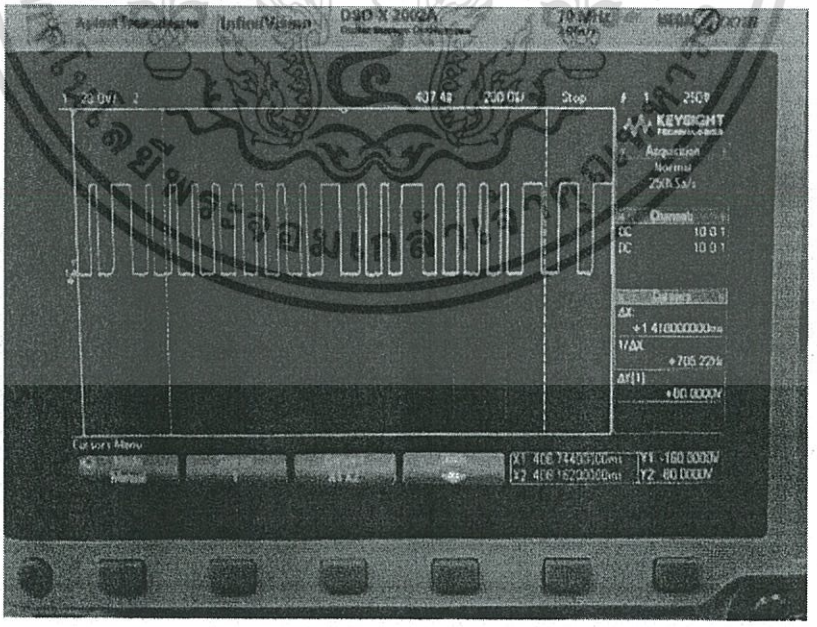
เป็นการอ่านค่าอุณหภูมิและความชื้นด้วย DHT22 แล้วนำมาแสดงบน Serial Monitor ของ Node MCU ค่าที่อ่านได้จาก DHT22 คือ ความชื้นมีค่าเป็น 45.00 % และ อุณหภูมิมีค่าเป็น 28.90

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.2 สัญญาณที่ได้จากการวัดขาสัญญาณของ DHT22

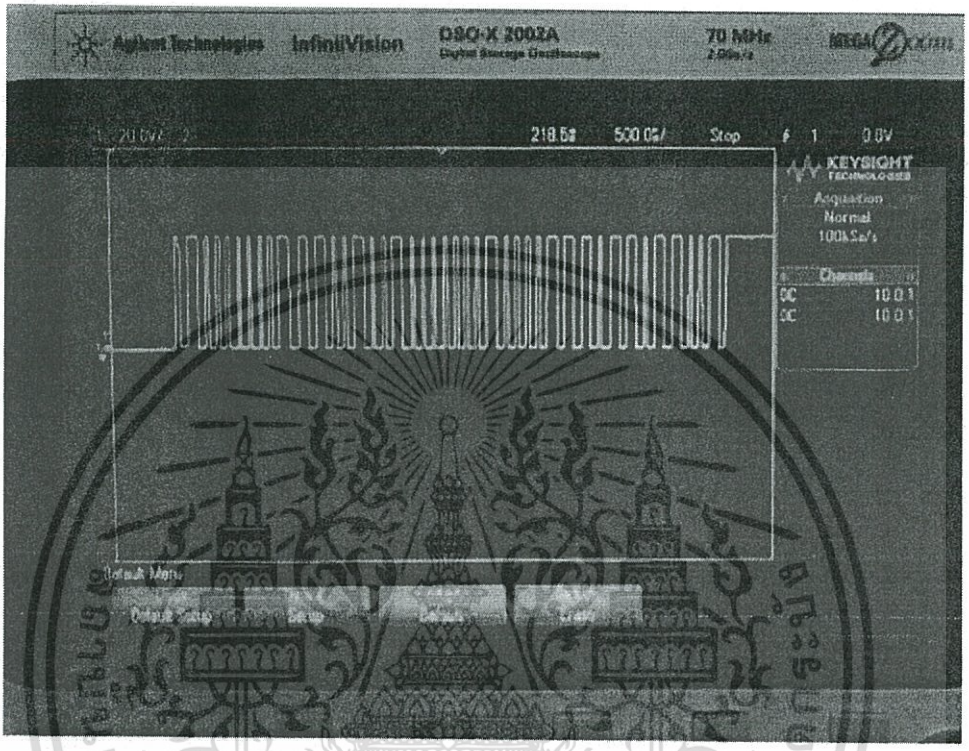
ทำการวัดสัญญาณจากขาสัญญาณของ DHT22 เพื่อนำมาอ่านค่าแล้วเปรียบเทียบกับค่าที่อ่านได้ใน Serial Monitor จาก datasheet ของ DHT22 บิตข้อมูลของอุณหภูมินั้นจะมี 16 บิต ถัดมาเป็นบิตข้อมูลของความชื้นอีก 16 บิต และ พาริตี้บิตอีก 8 บิต รวมทั้งหมดมี 40 บิต



รูปที่ 4.3 แสดงสัญญาณ 16 ของค่าอุณหภูมิ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

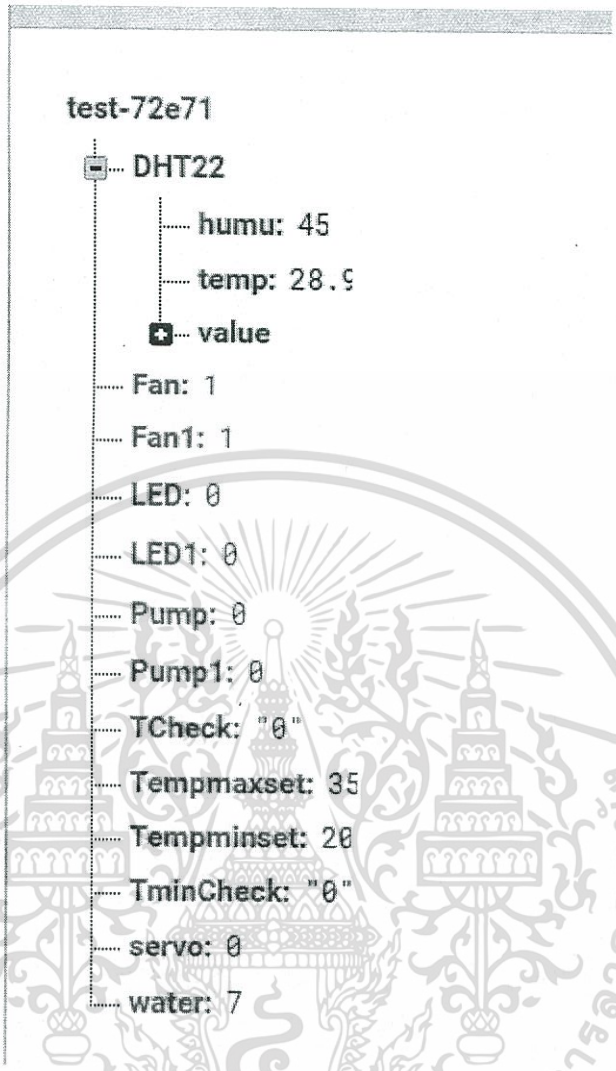
โดยเมื่อตัด 8 บิตท้ายสุดที่เป็น Parity bit ออกแล้วนับไป 16 บิตจะได้เป็นบิตที่ใช้ในการอ่านค่าอุณหภูมิออกมา ทำการอ่านค่าบิตดังกล่าวได้เป็น 0000 0001 0010 0001 หรือ 289 ได้ค่าอุณหภูมิเป็น 28.9 C ซึ่งตรงกับ Serial Monitor



รูปที่ 4.4 สัญญาณ 16 ของค่าความชื้น

หลังจากอ่านค่าอุณหภูมิ 16 บิตไปแล้วนั้นถัดขึ้นมาอีก 16 บิตเป็นบิตข้อมูลของค่าความชื้น ทำการอ่านค่าบิตดังกล่าวได้เป็น 0000 0001 1100 0010 หรือ 450 ได้ค่าความชื้นเป็น 45.0 %RH ซึ่งตรงตามที่วัดได้จาก Serial Monitor

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



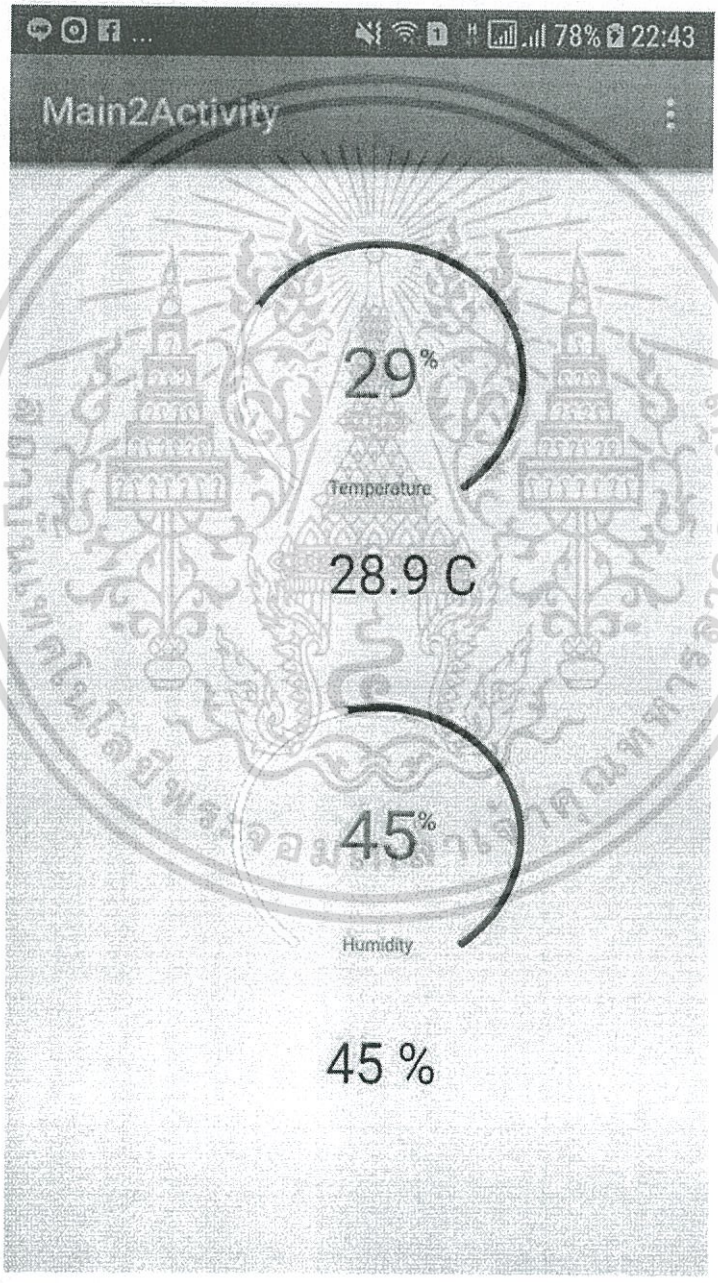
รูปที่ 4.5 ค่าอุณหภูมิและความชื้นบน Realtime database ของ Firebase

หลังจากที่ได้ค่าอุณหภูมิและความชื้นของ DHT22 แล้วจะทำการส่งค่าที่ได้ไปเก็บที่ humu คือ ค่าความชื้นซึ่งมีค่าเท่ากับ 45 และ temp คือ ค่าอุณหภูมิซึ่งมีค่าเท่ากับ 28.9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 การทดสอบการแสดงผลค่าอุณหภูมิบน Smart phone

ในการแสดงผลค่าอุณหภูมิและความชื้นบน Smart phone นั้นได้มาจากการอ่านค่าอุณหภูมิและความชื้นบน Realtime database ของ Firebase ที่ได้มาจาก Node MCU ESP826ส่งขึ้นมาบน Realtime database ของ Firebase



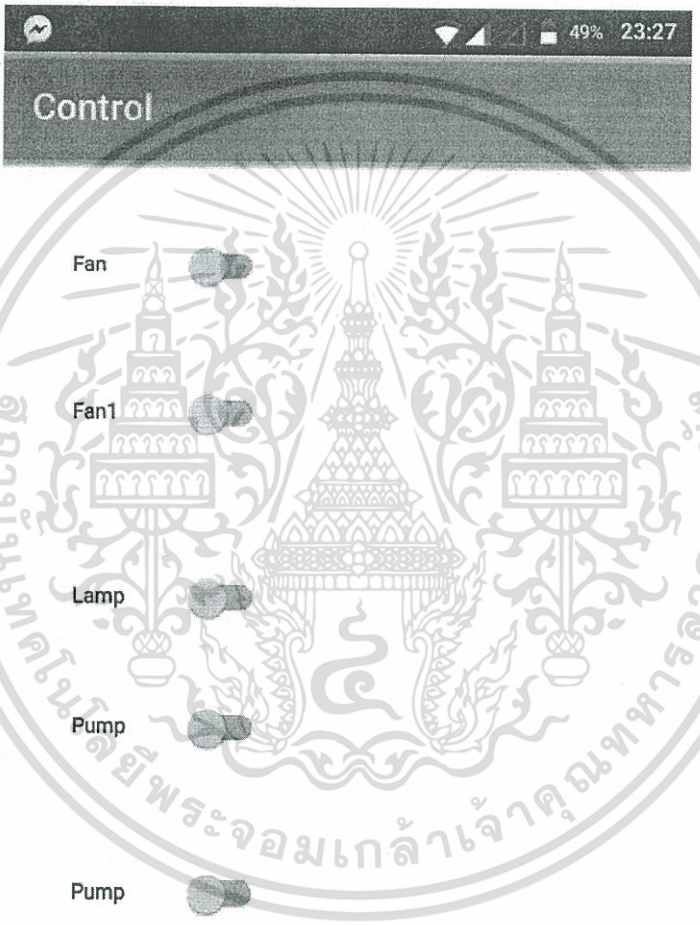
รูปที่ 4.6 ค่าอุณหภูมิบน Smart phone

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อค่าอุณหภูมิและความชื้นถูกส่งขึ้น Firebase ด้วย Node MCU ESP8266 แอปพลิเคชันจะทำการอ่านค่าดังกล่าวแล้วมาแสดงบนหน้าจอ

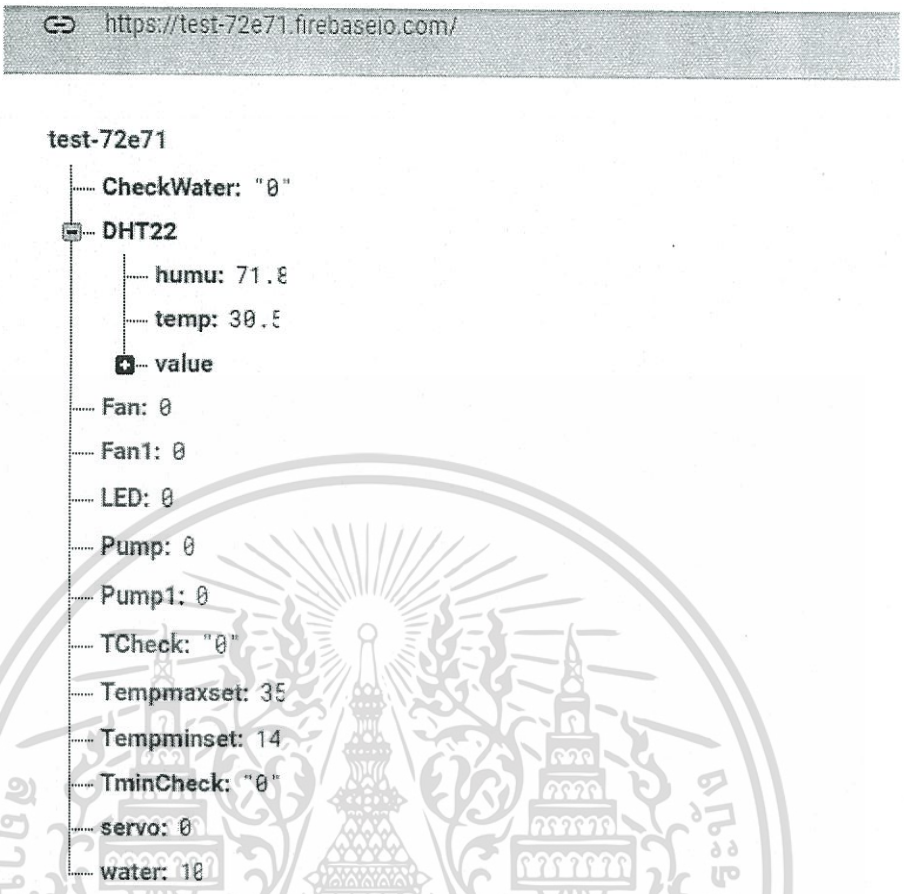
4.3 การทดสอบการเปิด-ปิดด้วยแอปพลิเคชัน

ในการเปิดและปิดจะใช้การส่งค่าคำสั่งไปเก็บบน Realtime database ของ Firebase เมื่อมีการกดปุ่มแล้ว Node MCU ESP8266 อ่านค่าดังกล่าวมาตัดสินการทำงานว่าให้เปิดและปิดอุปกรณ์ตัวใดตามปุ่มที่กด



รูปที่ 4.7 หน้าแอปพลิเคชันในการควบคุมการเปิดปิด

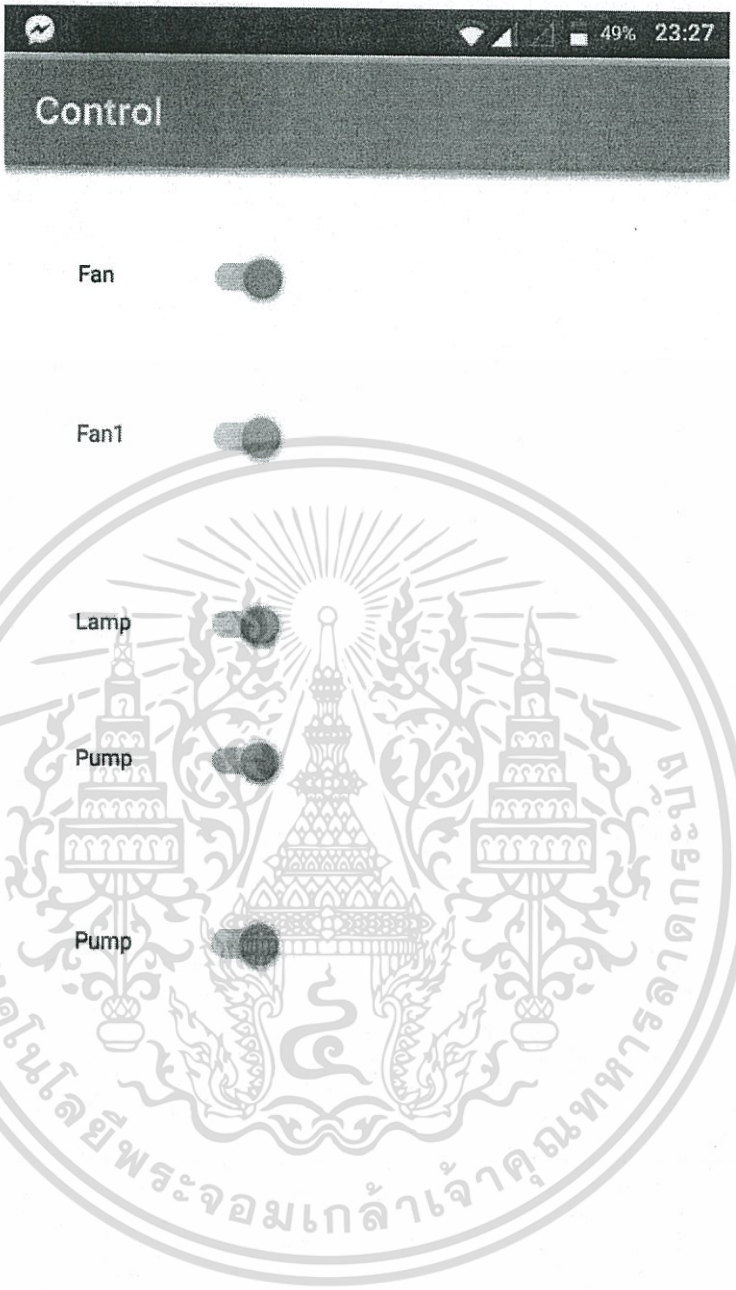
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.8 ค่าที่เก็บใน firebase ตอนยังไม่กดปุ่ม

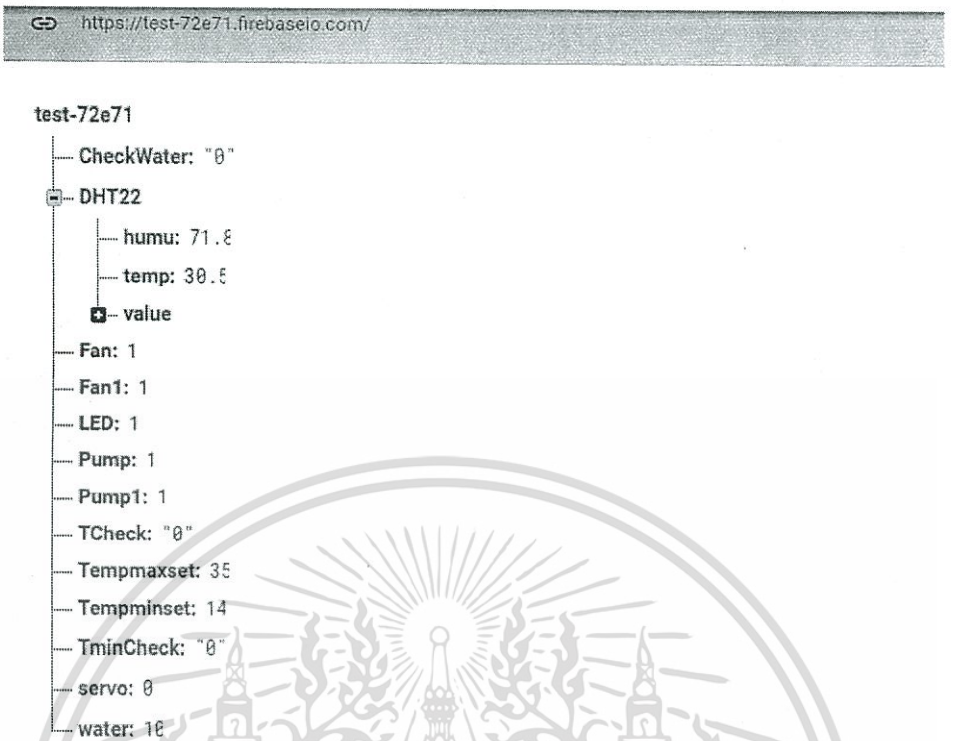
เมื่อยังไม่กดปุ่มเปิดค่าบน Firebase จะเป็น 0 แล้วรีเลย์จะไม่ทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



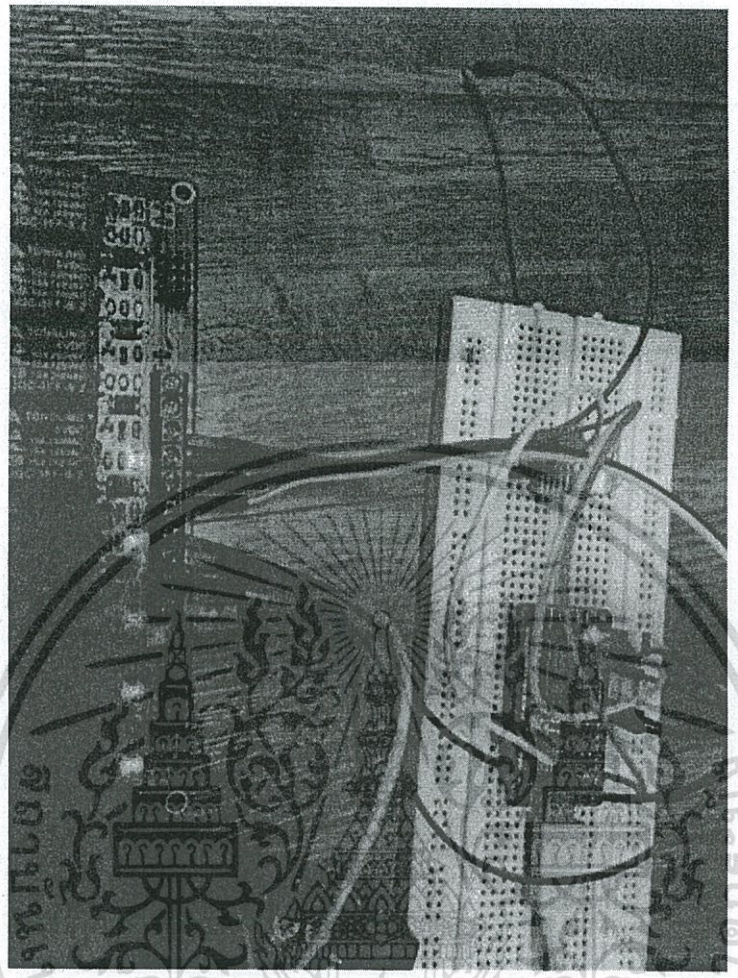
รูปที่ 4.9 แสดงหน้าแอปพลิเคชันเมื่อกดเปิด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.10 ค่าที่เก็บใน firebase เมื่อกดปุ่มเปิด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



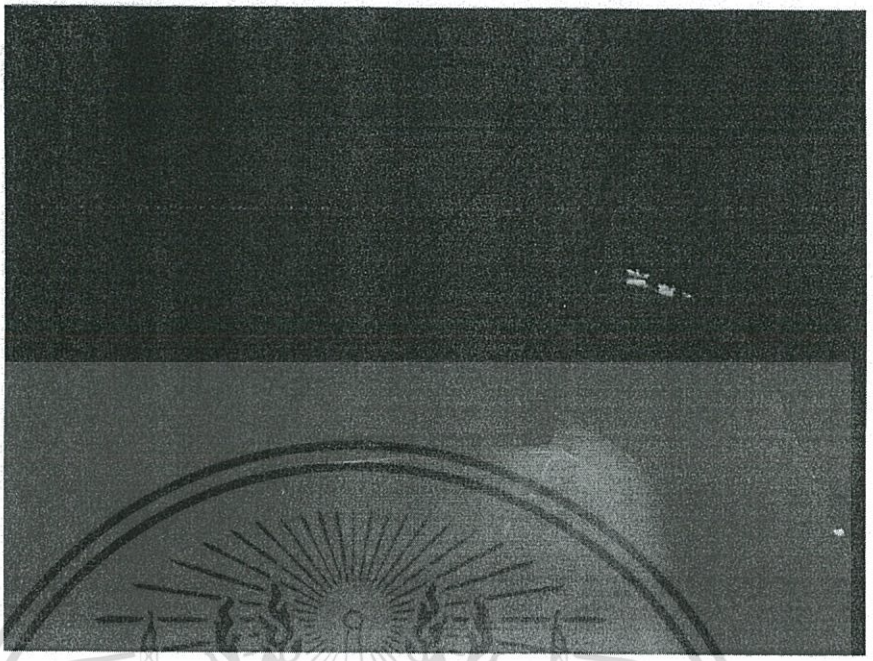
รูปที่ 4.11 เมื่อกดปุ่มเปิดรีเลย์จะทำงาน

เมื่อกดปุ่มเปิดค่าบน Firebase จะเป็น 1 แล้วรีเลย์จะทำงาน

4.4 การทดสอบการวัดค่าระดับน้ำ

ในการวัดค่าระดับน้ำนั้นจะใช้ ultrasonic hc-sr04 โดยการวัดค่าความสูงจากระดับน้ำถึงตัว ultrasonic hc-sr04 จากนั้นนำค่าที่วัดได้มาลบกับความสูงของกล่องที่ใส่น้ำจะได้มาเป็นความสูงของน้ำจากพื้นของกล่องใส่น้ำจากนั้นนำค่าที่ได้ส่งไปเก็บบน Realtime database ของ Firebase

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.12 ระดับของน้ำที่จะทำการวัด

COM3

```
Led: 0
Fan: 0
Fan1: 0
Pump: 0
Pump1: 0
servo: 0
Tempset: 35
Firebase Pushed /DHT22/value -L8cL_bM1Ssx21MixlzI
Humidity: 72.30 %       Temperature: 30.40 *C
5 cm


Led: 0
Fan: 0
Fan1: 0
Pump: 0
Pump1: 0
servo: 0
Tempset: 35
Firebase Pushed /DHT22/value -L8cLc1NDpBABYDX5HLz
Humidity: 72.30 %       Temperature: 30.40 *C
5 cm


Led: 0
Fan: 0
```

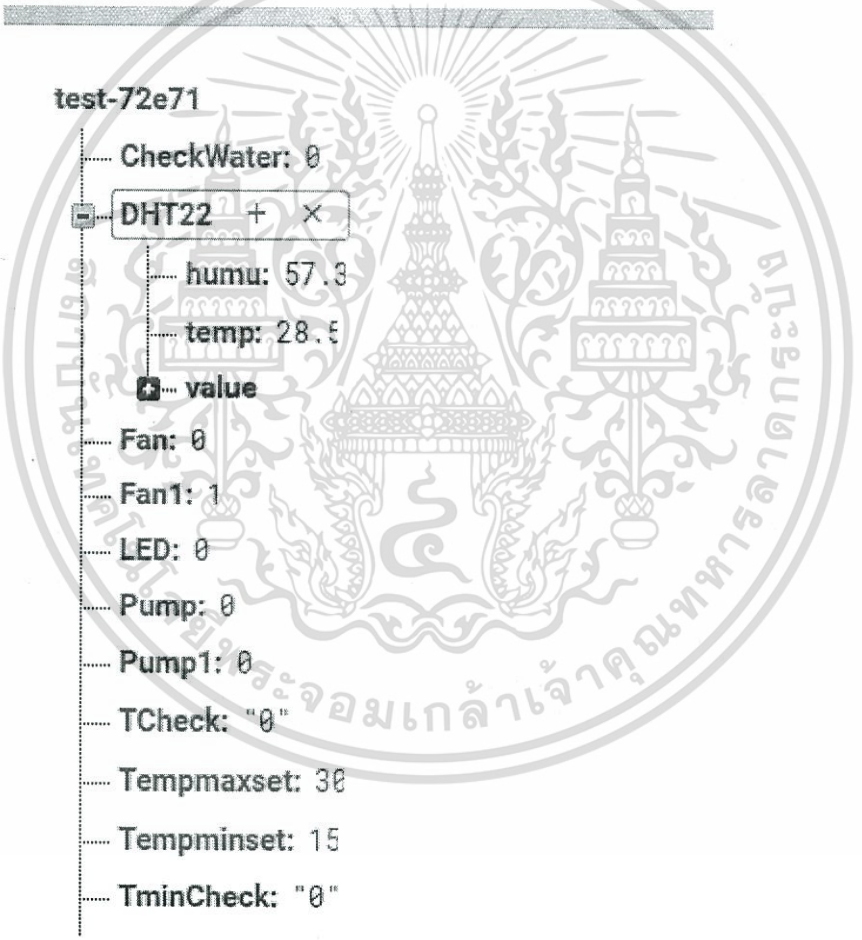
รูปที่ 4.13 ระดับของน้ำใน serial monitor จากเซนเซอร์ hc-sr04

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทำการวัดระดับน้ำก่อนได้ค่าเป็น 5 เซนติเมตร จากนั้นทดลองวัดด้วย HC SR04 ค่าที่ได้จาก Serial Monitor คือ 5 เซนติเมตร ซึ่งมีค่าตรงตามที่วัดไว้

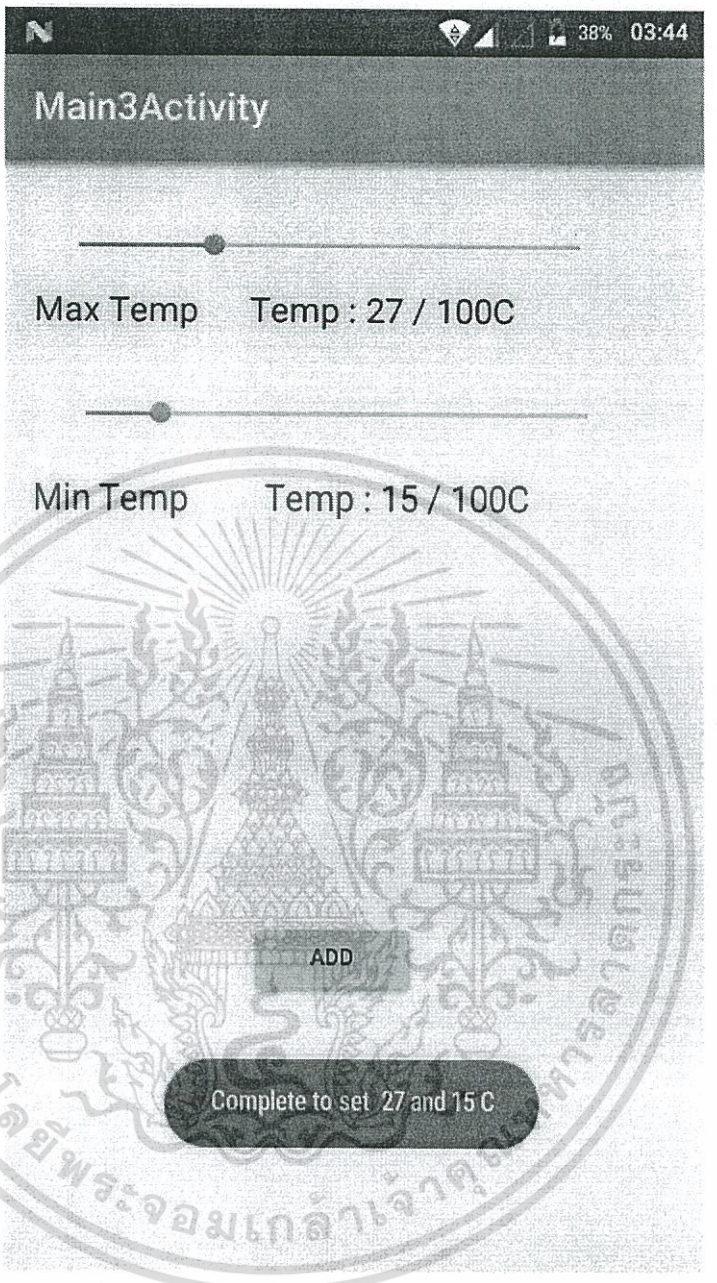
4.5 การทดสอบการแจ้งเตือน

การแจ้งเตือนนั้นจะอาศัยการตัดสินใจของ Node MCU ESP8266 และส่งผลที่ได้มาเก็บบน Realtime database ของ Firebase จากนั้นแอปพลิเคชันจะอ่านค่านั้นมาแล้วเพื่อตัดสินใจว่าต้องทำการแจ้งเตือนหรือไม่



รูปที่ 4.14 ค่าเริ่มต้นของอุณหภูมิที่กำหนด

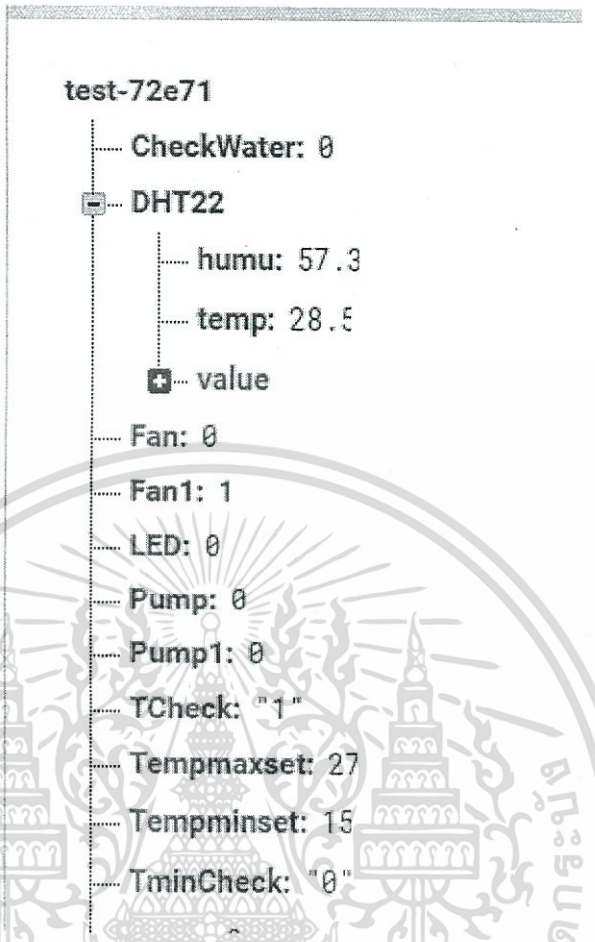
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.15 การกำหนดค่าอุณหภูมิในแอปพลิเคชัน

จากรูป 4.15 เป็นการตั้งค่าผ่านแอปพลิเคชันโดยเซ็ตค่าอุณหภูมิสูงสุดไว้ที่ 27 และตั้งค่าอุณหภูมิต่ำสุดไว้ที่ 15 แล้วกดส่งค่าขึ้น Firebase

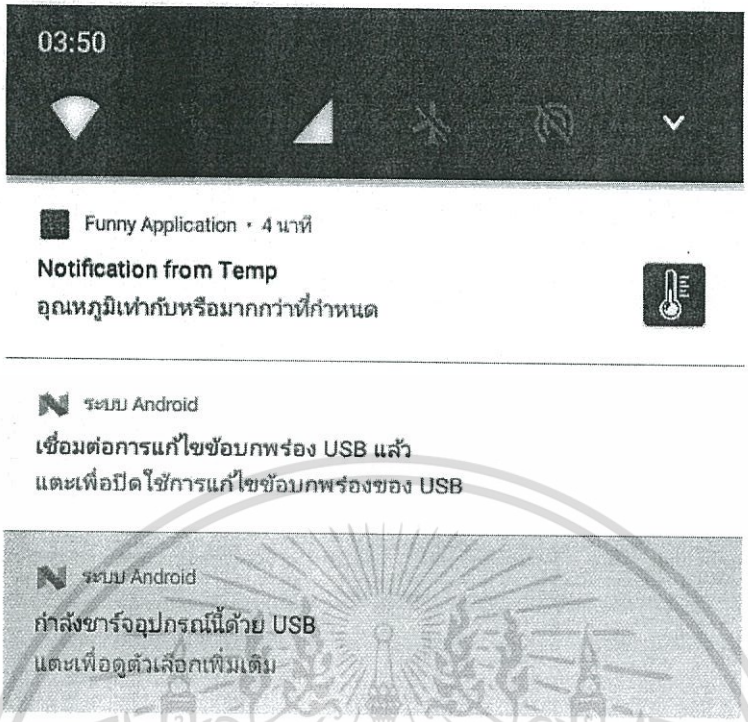
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.16 ค่าที่เก็บใน firebase หลังจากตั้งค่าอุณหภูมิในแอปพลิเคชัน

จากรูป 4.16 เมื่อกดยืนยันการตั้งค่าแล้วค่าจะถูกส่งมายัง Firebase โดยที่ค่าอุณหภูมิสูงสุดจะถูกเก็บไว้ที่ Tempmaxset และ ค่าอุณหภูมิต่ำสุดจะถูกเก็บไว้ที่ Tempminset Node MCU ESP8266 จะทำการอ่านดังกล่าวและตัดสินใจค่าอุณหภูมีที่วัดได้มีค่ามากกว่าค่า Tempmaxset จะทำการส่งค่า TCheck เท่ากับ 1 มายัง Firebase

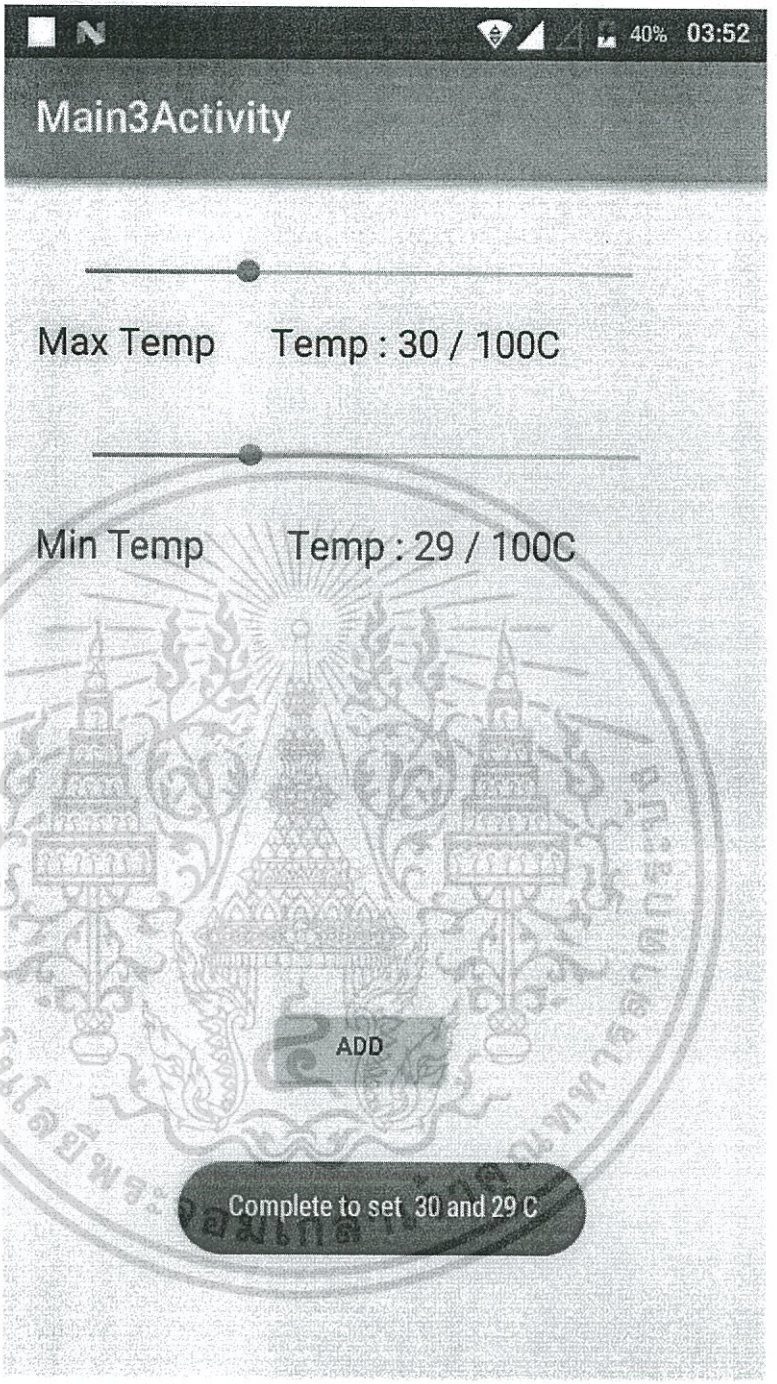
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.17 การแจ้งเตือนเมื่ออุณหภูมิที่ตั้งค่าไว้ต่ำกว่าอุณหภูมิที่อ่านได้

จากรูป 4.17 แอปพลิเคชันจะทำการอ่านค่าจาก Firebase ว่า TCheck เป็น 1 หรือไม่ถ้าใช่จะทำการแจ้งเตือนขึ้นมา ว่าอุณหภูมิมากกว่าหรือเท่ากับที่กำหนด

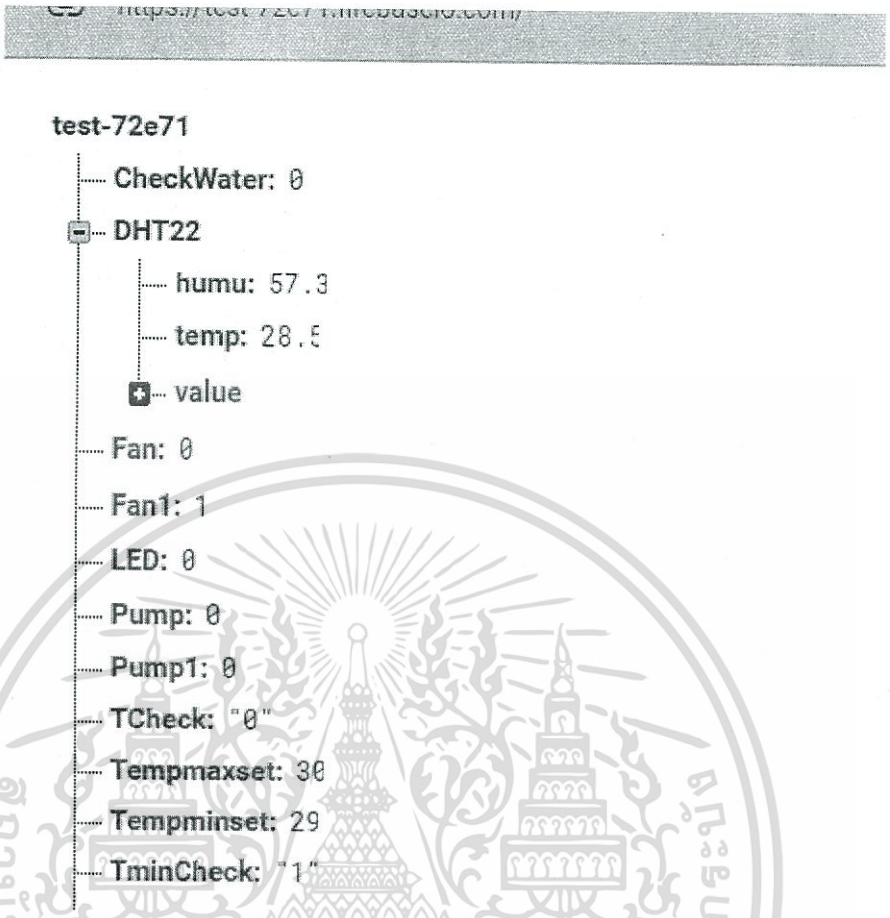
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.18 การตั้งค่าอุณหภูมิต่ำสุดให้มีค่ามากกว่าอุณหภูมิที่อ่านได้

จากรูป 4.18 ทำการตั้งค่าอุณหภูมิสูงสุดเป็น 30 และ อุณหภูมิต่ำสุดเป็น 29 แล้วทำการกดยืนยันค่าจะถูกส่งมาเก็บบน Firebase

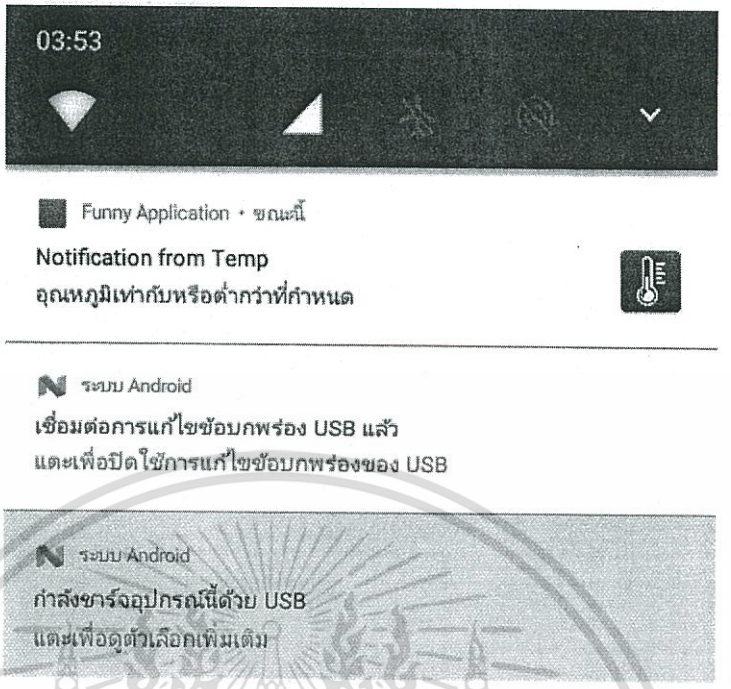
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.19 ค่าที่เก็บใน firebase หลังจากการตั้งค่าอุณหภูมิ

จากรูป 4.19 เมื่อกดยืนยันการตั้งค่าแล้วค่าจะถูกส่งมายัง Firebase โดยที่ค่าอุณหภูมิสูงสุดจะถูกเก็บไว้ที่ Tempmaxset และ ค่าอุณหภูมิต่ำสุดจะถูกเก็บไว้ที่ Tempminset Node MCU ESP8266 จะทำการอ่านดังกล่าวและตัดสินใจค่าอุณหภูมีที่วัดได้มีค่าน้อยกว่าค่า Tempminset จะทำการส่งค่า TminCheck เท่ากับ 1 มายัง Firebase

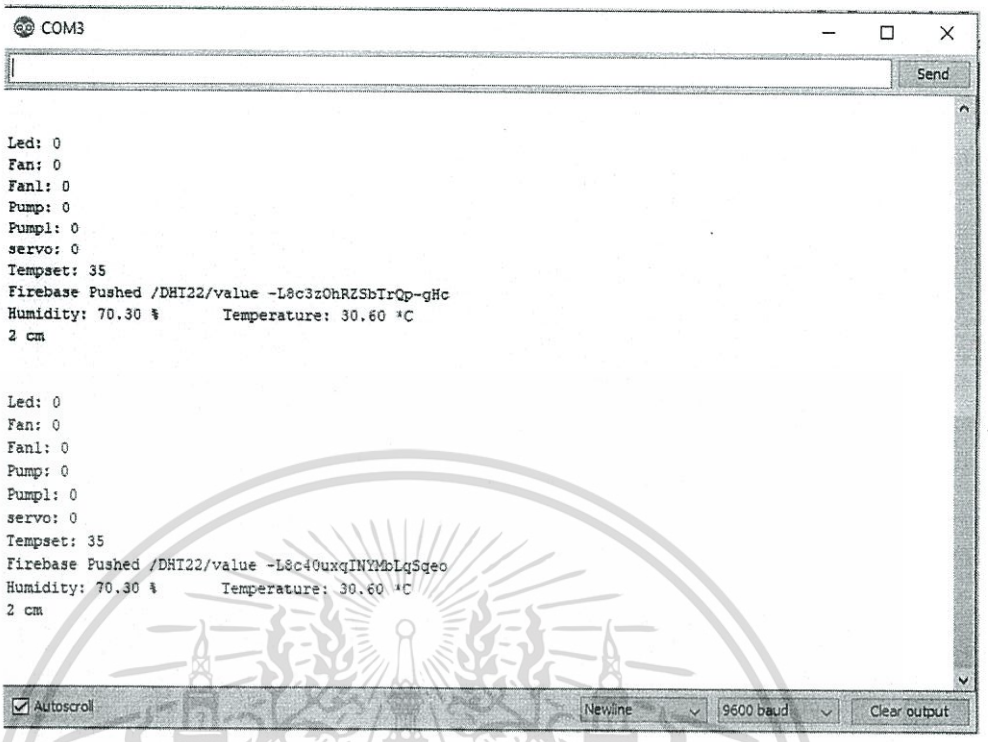
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.20 การแจ้งเตือนเมื่ออุณหภูมิที่ตั้งค่าไว้มากกว่าอุณหภูมิที่อ่านได้

จากรูป 4.20 แอปพลิเคชันจะทำการอ่านค่าจาก Firebase ว่า TminCheck เป็น 1 หรือไม่ถ้าใช่จะทำการแจ้งเตือนขึ้นมา ว่าอุณหภูมิน้อยกว่าหรือเท่ากับที่กำหนด

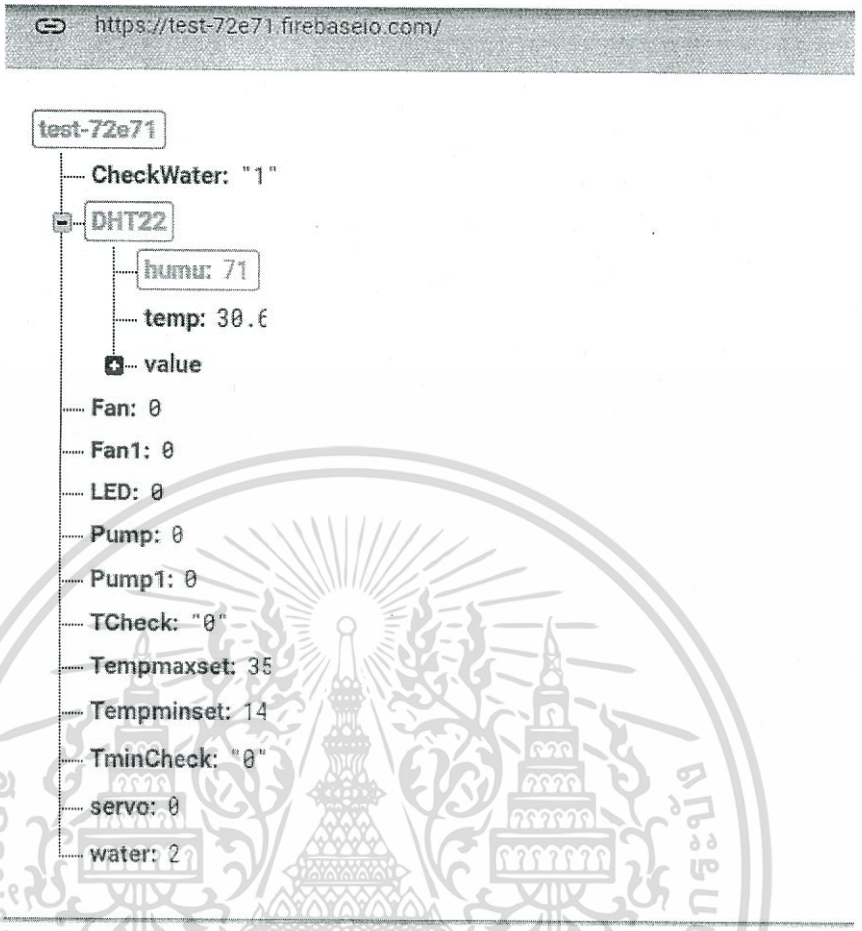
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.21 ค่าบน Serial Monitor ของ Node MCU ESP8266

จากรูป 4.21 แสดงค่าระดับน้ำเมื่อต่ำกว่าเกณฑ์บน Serial Monitor โดยวัดจาก HC-SR04 ได้ 2 เซ็นติเมตร

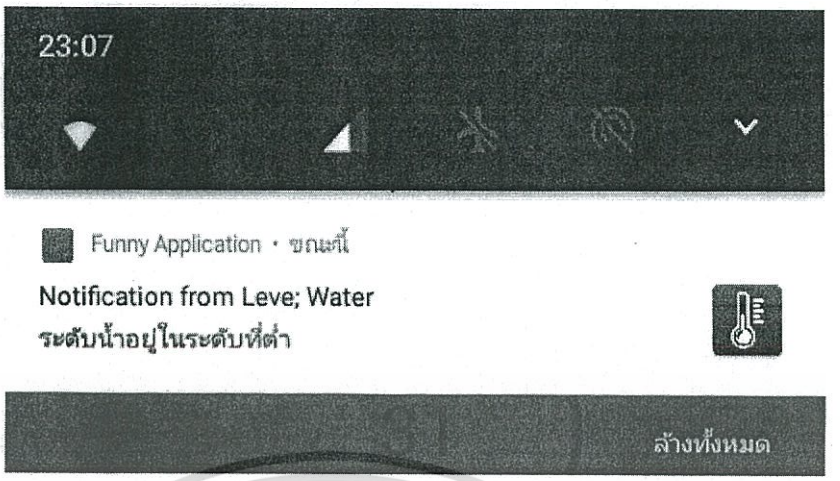
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.22 การค่าที่เก็บใน firebase หลังจากวัดระดับน้ำ

จากรูป 4.22 เมื่อระดับน้ำต่ำกว่าที่ตั้งไว้ Node MCU ESP8266 จะทำการส่งค่า CheckWater เท่ากับ 1 มายัง Firebase

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.23 การแจ้งเตือนของแอปพลิเคชันเมื่อระดับน้ำต่ำกว่าเกณฑ์

จากรูป 4.23 แอปพลิเคชันทำการอ่านค่าบน Firebase แล้วค่า CheckWater เท่ากับ 1 จะทำการแจ้งเตือนว่าระดับน้ำอยู่ในระดับที่ต่ำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปผลและข้อเสนอแนะ

5.1 สรุปผล

โครงงานนี้มีการออกแบบการควบคุมแบ่งออกเป็น 2 ส่วน (1) ส่วนของชิ้นงานที่ใช้ควบคุมการทำงาน เช่น พัดลม หลอดไฟ ปั้มน้ำ เป็นต้น และมีการวัดค่าอุณหภูมิกับความชื้นโดย DHT22 ซึ่งจะอาศัย Realtime database ของ Firebase ทั้งในการอ่านค่าและบันทึกค่าลงไปบน Realtime database โดยการสั่งงานการควบคุมนั้นจะใช้การอ่านค่าตัวแปรนั้นๆบน Realtime database เพื่อนำมาสั่งการอีกที รวมทั้งยังใช้เพื่อเป็นการบันค่าอุณหภูมิกับความชื้น และ ค่าต่างๆเพื่อนำไปใช้ในภายหลังอีกด้วย (2) ส่วนของแอพพลิเคชั่นก็จะมีการแสดงค่าอุณหภูมิและความชื้นโดยอาศัยการอ่านค่าอุณหภูมิและความชื้นที่บันทึกมาจาก Node MCU ESP8266 ที่รับค่ามาจากDHT22 แล้วบันทึกมายัง Realtime database ในหน้าการควบคุมนั้นจะสามารถสั่งการเปิดปิดได้ โดยเมื่อทำการกดปุ่มเปิดหรือปิดจะทำการบันทึกค่าลงไปบน Realtime database แล้วให้ Node MCU ESP8266 ทำการอ่านค่าเพื่อนำมาสั่งการต่อ สุดท้ายเป็นการแจ้งเตือนบนแอพพลิเคชั่น จะมีสองรูปแบบคือ ค่าอุณหภูมิกับระดับน้ำ เมื่อทำการตั้งค่าอุณหภูมิแล้วแอพพลิเคชั่นจะทำการแจ้งเตือน เช่นเดียวกับระดับน้ำที่เมื่อระดับน้ำถึงเกณฑ์ที่ตั้งไว้แอพพลิเคชั่นจะทำการแจ้งเตือนอัตโนมัติ

5.2 ข้อเสนอแนะ

ในระบบยังพบข้อบกพร่องเรื่องระยะเวลาในการทำงานที่ต้องทำผ่านการอ่านค่าบน Realtime database ของ Firebase ซึ่งจะเกิดความล่าช้าในการทำงานส่งผลให้ระบบไม่ทำงานในทันทีจึงจำเป็นต้องรอสักระยะหนึ่งก่อนระบบถึงจะทำงานตามที่สั่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

[1] “โรงเรือนระบบ Evap”

http://www.ppf-poultry.com/index.php?lay=show&ac=article&Id=539339728

[2] “Arduino Startup”

https://www.gravitechthai.com/guru2.php?p=260

[3] “รีเลย์ (Relay)”

http://www.psptech.co.th/%E0%B8%A3%E0%B8%B5%E0%B9%80%E0%B8%A.page

[4] “ESP8266 NodeMCU”

https://embeddedsystem2558.wordpress.com/esp8266-nodemcu-%E0%B8%84%E0%B8%B7%E0%B8%AD%E0%B8%AD%E0%B8%B0%E0%B9%84%E0%B8%A3-%E0%B9%81%E0%B8%A5%E0%B8%B0%E0%B8%81%E0%B8%B2%E0%B8%A3%E0%B8%95%E0%B8%B4%E0%B8%94%E0%B8%95%E0%B8%B1%E0%B9%89%E0%B8%87-e/

[6] “DHT22”

https://embed58.learninginventions.org/%E0%B8%AB%E0%B8%B1%E0%B8%A7%E0%B8%82%E0%B9%89%E0%B8%AD%E0%B8%99%E0%B8%B3%E0%B9%80%E0%B8%AA%E0%B8%99%E0%B8%AD/g1-4/

[7] “RC Servo Motor”

http://www.thaieasyelec.com/article-wiki/review-product-article/%E0%B8%9A%E0%B8%97%E0%B8%84%E0%B8%A7%E0%B8%B2%E0%B8%A1%E0%B8%95%E0%B8%B1%E0%B8%A7%E0%B8%AD%E0%B8%A2%E0%B9%88%E0%B8%B2%E0%B8%87%E0%B8%81%E0%B8%B2%E0%B8%A3%E0%B8%84%E0%B8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

%A7%E0%B8%9A%E0%B8%84%E0%B8%B8%E0%B8%A1-rc-servo-motor-%E0%B8%94%E0%B9%89%E0%B8%A7%E0%B8%A2-arduino.html

[8] "Firebase"

http://www.softmelt.com/article.php?id=588

[9] "Android studio"

http://www.thaicreate.com/mobile/android-studio-ide.html

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้