ระบบควบคุมตู้ปลาอัจฉริยะ

(Smart Fish Tank Controller)

กำพล อู่นาค

GUMPOL AUNAK

อนาวิล ศิริวรรณา

ANAWIN SIRIWANNA

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ.2565

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบควบคุมตู้ปลาอัจฉริยะ

(Smart Fish Tank Controller)

โดย

กำพล อู่นาค รหัสนักศึกษา 62010061

อนาวิล ศิริวรรณา รหัสนักศึกษา 62011010

อาจารย์ที่ปรึกษา

ผศ.ดร.วีระ เพ็งจันทร์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ.2565

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2565

ภาควิชา วิศวกรรมอิเล็กทรอนิกส์

คณะ วิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ระบบควบคุมตู้ปลาอัจฉริยะ

(Smart Fish Tank Controller)

ผู้จัดทำ นาย กำพล อู่นาค รหัสประจำตัว 62010061

นาย อนาวิล ศิริวรรณา รหัสประจำตัว 62011010

ปริญญานิพนธ์นี้ผ่านการตรวจสอบโดยอาจารย์ที่ปรึกษาแล้ว

(ผศ.ดร.วีระ เพ็งจันทร์)

อาจารย์ที่ปรึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อโครงงาน	ระบบควบคุมตู้ปลาอัจฉริยะ	
นักศึกษา	นาย กำพล อู่นาค	รหัสนักศึกษา 62010061
	นาย อนาวิล ศิริวรรณา	รหัสนักศึกษา 62011010
ปริญญา	วิศวกรรมศาสตรบัณฑิต	
ภาควิชา	วิศวกรรมอิเล็กทรอนิกส์	
ปีการศึกษา	2565	
อาจารย์ที่ปรึกษาโครงการ	ผศ.ดร.วีระ เพ็งจันทร์	

บทคัดย่อ

รายงานฉบับนี้เป็นส่วนหนึ่งของวิชา PROJECT2 มีเนื้อหาเกี่ยวกับการออกแบบระบบควบคุมตู้ปลาอัจฉริยะ (Smart Fish Tank Controller) มีวัตถุประสงค์เพื่อเป็นการฝึกการใช้งานไมโครคอนโทรลเลอร์และเซ็นเซอร์ทั้งทางด้านฮาร์ดแวร์และเฟิร์มแวร์ โดยส่วนของไมโครคอนโทรลเลอร์ควบคุมการทำงานจะใช้ ESP32 DEVKIT V1 และใช้โปรแกรม Arduino IDE ในการเบิร์นโค้ด ส่วนของการควบคุมอุปกรณ์เลี้ยงปลาต่างๆจะใช้เซ็นเซอร์เช็คค่าพารามิเตอร์ที่สำคัญเกี่ยวกับเลี้ยงปลา คือ อุณหภูมิน้ำ ความเป็นกรดด่างในน้ำ และความสะอาดของน้ำ แล้วใช้รีเลย์ในการควบคุมการทำงานของอุปกรณ์ คือ สั่งให้ฮีตเตอร์และพัดลมทำงาน สั่งให้ปั๊มกรดด่างทำงาน และสั่งให้ปั๊มน้ำทำงาน เป็นต้น นอกจากนี้ระบบควบคุมตู้ปลาอัจฉริยะนี้ยังสามารถควบคุมผ่านจอระบบสัมผัสที่กล่องควบคุมได้ สำหรับการทำงานของระบบควบคุมตู้ปลาอัจฉริยะจะมีโหมดการทำงานอยู่ 2 โหมด คือ โหมด Manual สั่งให้อุปกรณ์ทำงานด้วยตัวเราเอง และโหมด Auto ที่อุปกรณ์จะทำงานเมื่อค่าพารามิเตอร์สำหรับการเลี้ยงปลาที่เรากำหนดไว้มีค่าต่ำ/สูงกว่าเกณฑ์ที่เรากำหนด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Project Title	Smart Fish Tank Controller	
Student	Mr. Gumpol Aunak	Student ID 62010061
	Mr. Anawin Siriwanna	Student ID 62011010
Degree	Bachelor of Engineering	
Program	Electronics Engineering	
Year	2022	
Project Advisor	Asst. Prof. Dr. Weera Pengchan	

ABSTRACT

This report is a part of PROJECT2 subject about the design of Smart aquarium control system (Smart Fish Tank Controller) aims to train the use of microcontrollers and sensors in both hardware and firmware. The microcontroller part uses the ESP32 DEVKIT V1 and uses the Arduino IDE program to burn the code. As for the control of various fish farming equipment, sensors are used to check important parameters related to fish farming, namely water temperature, pH in the water. and cleanliness of the water. Then use the relays to control the operation of the device, that is, order the heater and fan to work. Command the acid-base pump to work. And instruct the water pump to work, etc. In addition, this intelligent fish tank control system can also be controlled via the touch screen on the control box. For the operation of the intelligent aquarium control system, there are 2 modes of operation: Manual mode, ordering the device to work by ourselves and Auto mode that the device will work when the parameters for fish farming that we set are low / higher than the threshold we set.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

การทำโครงงานในครั้งนี้ต้องขอขอบคุณ อาจารย์ที่ปรึกษา ผศ.ดร.วีระ เพ็งจันทร์ ที่คอยชี้แนะและให้คำปรึกษาในการเรื่องต่างๆ ไม่ว่าจะเป็นการใช้อุปกรณ์ การเลือกซื้อของ ทฤษฎีความรู้ต่างๆ ขอบคุณพ่อแม่ที่คอยให้กำลังใจและพร้อมที่จะช่วยเหลือเมื่อพวกเราเจอปัญหา ขอบคุณบุคลากรภาควิชาวิศวกรรมอิเล็กทรอนิกส์ คณะวิศวกรรมศาสตร์ ที่ให้ใช้สถานที่และให้คำปรึกษาด้านงบประมาณ ขอบคุณเพื่อนๆที่คอยให้ความช่วยเหลือและให้กำลังใจ

กำพล อู่นาค

อนาวิล ศิริวรรณา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้าที่

บทคัดย่อ i

ABSTRACT ii

กิตติกรรมประกาศ iii

บทที่ 1 1

บทนำ 1

1.1 ที่มาและความสำคัญ 1

1.2 วัตถุประสงค์ 2

1.3 ขอบเขตการศึกษา 2

1.4 ประโยชน์ที่คาดว่าจะได้รับ 2

1.5 ระยะเวลาในการทำโครงงาน 3

บทที่ 2 4

ทฤษฎีอุปกรณ์ 4

2.1 ไมโครคอนโทรลเลอร์ 4

2.2 ESP32 8

2.3 ESP32 DEVKIT V1 21

2.4 Arduino IDE 22

2.5 DS18B20 Temperature Sensor 24

2.6 pH4502 รุ่น E201-C 25

2.7 KS0429 keyestudio TDS Meter V1.0 26

2.8 Nextion 27

2.9 Relay module 28

2.10 Motor 29

2.11 Firebase 30

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.12 Flutter 32
2.13 Dart 35
2.14 Android Studio 35
2.15 Visual Studio 36
2.16 Visual Studio Code 37
2.17 พัดลมตู้ปลา (Fan) 38
2.18 ฮีทเตอร์สำหรับตู้ปลา (Aquarium heater) 39
2.19 ไฟ LED (Strip LED Light) 40
2.20 ปั๊มน้ำ (Water pump) 40
2.21 เครื่องกรองน้ำ (Filter) 41
2.22 ความสำคัญของน้ำต่อการดำรงชีพของปลา 42
2.23 น้ำที่ใช้เลี้ยงปลาสวยงาม 42
2.24 คุณสมบัติของน้ำที่เกี่ยวข้องกับการดำรงชีพของปลา 44
บทที่ 3 46
หลักการทำงานและการออกแบบ 46
3.1 อุปกรณ์ เครื่องมือ หรือโปรแกรมที่ใช้ในการดำเนินโครงงาน 46
3.2 ขั้นตอนการดำเนินงาน 47
3.3 บล็อกไดอะแกรม(Block Diagram) ผังงาน(Flow Chart) และวงจร(Schematic) 47
3.4 การใช้งานโปรแกรม Arduino IDE 51
3.5 การเชื่อมต่อและใช้งาน pH Sensor กับ ESP32 55
3.6 การเชื่อมต่อและใช้งาน Temperature Sensor กับ ESP32 57
3.7 การเชื่อมต่อและใช้งาน TDS Sensor กับ ESP32 60
3.8 การใช้งานจอ Nextion 2.8” Basic HMI Display 62
3.9 การแสดงค่า Sensor ทางจอ Nextion Display 66
3.10 การใช้จอ Nextion Display ควบคุมอุปกรณ์ 72

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.11 การส่งค่าไปยัง Firebase Realtime Database 75
3.12 การติดตั้งซอฟต์แวร์สำหรับการพัฒนาแอพพลิเคชั่น 85
3.13 พัฒนาแอพพลิเคชั่น 90
3.14 การส่งข้อมูลจาก Firebase มาที่แอพพลิเคชั่นของ Flutter 93
3.15 การประกอบและรวมอุปกรณ์ต่างๆ 98
บทที่ 4 100
การทดลอง และผลการทดลอง 100
4.1 การทดลองวัดค่าจากเซ็นเซอร์มาแสดงผลบนจอภาพ 100
4.2 การทดลองการควบคุมอุปกรณ์แบบอัตโนมัติ 110
4.3 การทดลองวัดค่าจากจอภาพเทียบกับ Firebase 113
4.4 การทดลองวัดค่าจากแอพลิเคชั่นเทียบกับ Firebase 113
บทที่ 5 116
สรุปผลและข้อเสนอแนะ 116
5.1 สรุปผล 116
5.2 ข้อเสนอแนะ 116
บรรณานุกรม 117
ภาคผนวก 119

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ

หน้าที่

รูปที่ 2.1 โครงสร้างโดยทั่วไปของไมโครคอนโทรลเลอร์ 4
รูปที่ 2.2 สถาปัตยกรรมฟอนนอยมันน์ 6
รูปที่ 2.3 สถาปัตยกรรมฮาร์วาร์ด 7
รูปที่ 2.4 ชิป ESP32 8
รูปที่ 2.5 DOIT ESP32 Development Board 9
รูปที่ 2.6 โมดูล ESP-WROOM-32 10
รูปที่ 2.7 โครงสร้างของ ESP-WROOM-32 11
รูปที่ 2.8 วงจรของ ESP-WROOM-32 12
รูปที่ 2.9 หน้าตาของโมดูล ESP32S 12
รูปที่ 2.10 โครงสร้างของ ESP32-Bit (รูปซ้าย) และตำแหน่งขาต่อใช้งาน (รูปขวา) 13
รูปที่ 2.11 โมดูลรุ่น IntoRobot-W32 และรุ่น IntoRobot-W33 14
รูปที่ 2.12 ตำแหน่งขาต่าง ๆ และโครงสร้างของ IntoRobot-W32/W33 14
รูปที่ 2.13 บอร์ด ESP32-DevKitC เวอร์ชั่นแรก 15
รูปที่ 2.14 บอร์ด ESP32-DevKitC เวอร์ชั่นที่สอง เป็นเวอร์ชั่นที่จำหน่ายในปัจจุบัน 15
รูปที่ 2.15 ตำแหน่งขาบนบอร์ด ESP32-DevKitC 16
รูปที่ 2.16 บอร์ด SparkFun ESP32 Thing 17
รูปที่ 2.17 Pinout ของบอร์ด SparkFun ESP32 Thing 18
รูปที่ 2.18 บอร์ด NodeMCU-32S 19
รูปที่ 2.19 บอร์ด NodeMCU-32S ด้านหลัง 19
รูปที่ 2.20 Pinout ของบอร์ด NodeMCU-32S 20
รูปที่ 2.21 บอร์ด WEMOS LOLIN32 ด้านหน้าและด้านหลัง 21
รูปที่ 2.22 บอร์ด ESP32 DEVKIT V1 22
รูปที่ 2.23 บอร์ด ESP32 DEVKIT V1 pinout 22
รูปที่ 2.24 บอร์ด หน้าตาของโปรแกรม Arduino IDE แบบ desktop 23
รูปที่ 2.25 DS18B20 Temperature Sensor 24
รูปที่ 2.26 pH4502 รุ่น E201-C 25

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.27 ตารางค่าความเป็นกรดเบส 25
รูปที่ 2.28 KS0429 keyestudio TDS Meter V1.0 26
รูปที่ 2.29 ตารางค่า TDS ที่ยอมรับ 27
รูปที่ 2.30 จอ Nextion 27
รูปที่ 2.31 Relay module 28
รูปที่ 2.32 วงจรภายใน Relay module 29
รูปที่ 2.33 Dart 35
รูปที่ 2.34 Android studio 36
รูปที่ 2.35 Microsoft Visual Studio 37
รูปที่ 2.36 Visual Studio Code 38
รูปที่ 2.37 พัดลมตู้ปลา 38
รูปที่ 2.38 ฮีทเตอร์สำหรับตู้ปลา 39
รูปที่ 2.39 รูปภาพไฟ LED Strip 40
รูปที่ 2.40 รูปภาพปั๊มน้ำสำหรับตู้ปลา 41
รูปที่ 2.41 รูปภาพเครื่องกรองน้ำ 42
รูปที่ 3.1 รูปภาพแสดงบล็อกไดอะแกรม ของการทำงานต่างๆในส่วน input และ output ของไมโครคอนโทลเลอร์ 47
รูปที่ 3.2 รูปภาพผังงานแสดงการทำงานในโหมด Manual และ Auto 49
รูปที่ 3.3 รูปภาพวงจร แสดงการเชื่อมต่ออุปกรณ์ต่างๆเข้าด้วยกัน 50
รูปที่ 3.4 หน้าเว็บไซต์ของ Arduino 51
รูปที่ 3.5 หน้าในของแถบของ Software และปุ่ม Download 51
รูปที่ 3.6 รูปในหน้า Download 52
รูปที่ 3.7 รูป Icon ของตัวติดตั้ง Arduino IDE 52
รูปที่ 3.8 รูปในหน้าแถบ File 53
รูปที่ 3.9 รูปในหน้าต่าง Preference 53
รูปที่ 3.10 รูปในหน้าแถบ Tools 54
รูปที่ 3.11 รูปในหน้าต่าง Boards Manager 54
รูปที่ 3.12 รูปวงจรในส่วนของ pH sensor และ ESP32 55
รูปที่ 3.13 รูปภาพแสดงโค้ดการทำงานของ pH sensor 55
รูปที่ 3.14 รูปแสดงการเลือก Port ที่เชื่อมต่อ 56
รูปที่ 3.15 รูปภาพแสดงการอัปโหลดโปรแกรมลงบอร์ด 56
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.16 รูปภาพแสดงการเปิด Serial monitor 56
รูปที่ 3.17 รูปภาพแสดงการเลือก Buad rate และค่า pH ที่แสดงผ่าน Serial Monitor 57
รูปที่ 3.18 รูปวงจรในส่วนของ Temperature sensor และ ESP32 57
รูปที่ 3.19 รูปที่แสดงการเข้าหน้าต่าง Manage libraries 58
รูปที่ 3.20 รูปหน้าต่าง library Manager 58
รูปที่ 3.21 รูปภาพแสดงโค้ดการทำงานของ Temperature sensor 59
รูปที่ 3.22 รูปแสดงค่าอุณหภูมิที่แสดงผ่าน Serial Monitor 59
รูปที่ 3.23 รูปวงจรในส่วนของ TDS sensor และ ESP32 60
รูปที่ 3.24 รูปภาพแสดงโค้ดการทำงานของ TDS sensor 61
รูปที่ 3.25 รูปแสดงค่า TDS ที่แสดงผ่าน Serial Monitor 61
รูปที่ 3.26 รูปแสดงหน้าเว็บไซต์ของ Nextion 62
รูปที่ 3.27 รูปแสดงหน้า Download ของ Nextion 62
รูปที่ 3.28 รูป icon ของ Nextion edition ที่ดาวน์โหลดมา 63
รูปที่ 3.29 รูปแสดงโปรแกรม Nextion Editor 63
รูปที่ 3.30 รูปแสดงการสร้างโปรเจคใหม่ของโปรแกรม Nextion Editor 63
รูปที่ 3.31 รูปแสดงการเลือก Model และ ขนาดของจอที่ใช้งาน 64
รูปที่ 3.32 รูปแสดงการเลือกทิศทางของจอ 64
รูปที่ 3.33 รูปภาพแสดงหน้าต่างของโปรแกรมและหน้าต่างย่อยภายในโปรแกรม 65
รูปที่ 3.34 ภาพแสดงการออกแบบหน้า Dashboard ในโปรแกรม Photoshop 66
รูปที่ 3.35 ภาพแสดงการเพิ่มไฟล์รูปภาพในโปรแกรม Nextion Editor 66
รูปที่ 3.36 ภาพแสดงการเลือกรูปภาพมาใช้งาน 67
รูปที่ 3.37 ภาพที่แสดงการเพิ่มกล่องข้อความ 67
รูปที่ 3.38 ภาพของ Micro SD card และ Card Reader 68
รูปที่ 3.39 ภาพแสดงการ Save file ในโปรแกรม Nextion Editor 68
รูปที่ 3.40 หน้าต่าง TFT File output 69
รูปที่ 3.41 ภาพแสดงไฟล์ของหน้าจอ 69
รูปที่ 3.42 ภาพแสดงการใส่ SD card เพื่ออัพโหลดข้อมูลลงจอ Nextion 69
รูปที่ 3.43 รูปภาพแสดงการอัพโหลดไฟล์ของจอ Nextion 70
รูปที่ 3.44 รูปภาพของจอ Nextion ที่พร้อมใช้งาน 70
รูปที่ 3.45 โค้ดในส่วนที่ใช้ส่งค่าไปยังหน้าจอ Nextion 71

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.46 ภาพของจอที่แสดงค่าของ Sensor 71
รูปที่ 3.47 ภาพแสดงการเลือกใช้งานปุ่มกดในโปรแกรม Nextion Editor 72
รูปที่ 3. 48 รูปภาพแสดงหน้าจอพร้อมปุ่มที่ใช้ควบคุมอุปกรณ์ 72
รูปที่ 3.49 รูปภาพแสดงการติดตั้ง library ของ Nextion 73
รูปที่ 3.50 รูปภาพแสดงการประกาศตัวแปรของปุ่มที่ใช้ใน Nextion Editor 73
รูปที่ 3.51 รูปภาพของโค้ดที่สร้างฟังก์ชั่นของปุ่มต่างๆ 74
รูปที่ 3.52 ภาพในส่วนของโค้ดที่ทำการเรียกฟังก์ชั่นของปุ่มที่ใช้งาน 74
รูปที่ 3.53 ภาพในส่วนของการรับค่าจากปุ่ม 74
รูปที่ 3.54 ภาพการทดสอบการปิดและเปิดรีเลย์ผ่านหน้าจอสัมผัส 75
รูปที่ 3.55 รูปภาพแสดงการเข้าเมนู Manage Libraries 75
รูปที่ 3.56 รูปภาพแสดงการติดตั้ง library firebase ESP32 Client 76
รูปที่ 3.57 รูปภาพแสดงการ Include library ของ firebase 76
รูปที่ 3.58 รูปภาพแสดงการกำหนดค่าเริ่มต้นของ Internet และ Firebase 76
รูปที่ 3.59 รูปภาพแสดงตำแหน่งของปุ่ม Go to console 77
รูปที่ 3.60 รูปภาพแสดงหน้าสร้าง Project 77
รูปที่ 3.61 รูปภาพแสดงการตั้งชื่อโปรเจค 77
รูปที่ 3.62 รูปภาพแสดงการดำเนินการสร้างโปรเจค-v’it[[..... 78
รูปที่ 3.63 รูปภาพเมื่อระบบสร้างโปรเจคสำเร็จ 78
รูปที่ 3.64 รูปภาพแสดงการเข้าใช้งานในส่วน Realtime Database 79
รูปที่ 3.65 รูปภาพแสดงตำแหน่งของปุ่ม Create database 79
รูปที่ 3.66 รูปภาพแสดงขั้นตอนการ Set up database 80
รูปที่ 3.67 รูปภาพแสดงขั้นตอนการ Set up database ต่อ 80
รูปที่ 3.68 รูปภาพแสดงการเปลี่ยนคำสั่ง false เป็น true 81
รูปที่ 3.69 รูปภาพแสดงตำแหน่ง url ของ firebase 81
รูปที่ 3.70รูปภาพแสดงตำแหน่งของปุ่ม Users and permission 82
รูปที่ 3.71 รูปภาพแสดงตำแหน่งของ database Secret 82
รูปที่ 3.72 รูปภาพแสดงการใส่ API key และ Database url 83
รูปที่ 3.73 รูปภาพแสดงโค้ดการส่งค่าเซ็นเซอร์ไปที่ Firebase 83
รูปที่ 3.74 รูปภาพแสดงโค้ดการรับค่าจาก Firebase 83
รูปที่ 3.75 รูปภาพแสดงตำแหน่งปุ่มเพิ่ม path 84

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.76 รูปภาพแสดงหน้าต่างการปรับค่า Path 84
รูปที่ 3.77 รูปภาพแสดง path และ path ย่อยที่ได้ทำการสร้าง 85
รูปที่ 3.78 หน้าเว็บไซต์ของ Flutter 85
รูปที่ 3.79 ระบบปฏิบัติการณ์ของ Flutter 85
รูปที่ 3.80 ดาวน์โหลดและติดตั้ง Flutter 86
รูปที่ 3.81 ตั้งค่า PATH environment variable 86
รูปที่ 3.82 เช็คซอฟต์แวร์โดยใช้คำสั่ง flutter doctor 86
รูปที่ 3.83 หน้าเว็บไซต์ของ Android Studio 87
รูปที่ 3.84 ตั้งค่า PATH ของ Android Studio 87
รูปที่ 3.85 สร้าง Device จำลองการทำงานของแอพพลิเคชั่น 88
รูปที่ 3.86 ปรับขนาดหน้าจอของ Devices 88
รูปที่ 3 87 หน้าเว็บไซต์ของ Visual Studio Code 89
รูปที่ 3.88 Flutter Extension ใน Visual Studio Code 89
รูปที่ 3. 89 Dart Extension ใน Visual Studio Code 90
รูปที่ 3.90 สร้างโปรเจคต์ใหม่ใน Visual Studio Code 90
รูปที่ 3.91 โปรเจคต์ดั้งเดิมใน Flutter 90
รูปที่ 3.92 แอพดั้งเดิมใน Flutter เป็นแอพที่กดปุ่มบวกแล้วเลขตรงกลางจะเพิ่ม 91
รูปที่ 3.93 เลือก Device ที่เราต้องการ 91
รูปที่ 3.94 ตัวอย่างแอพพลิเคชั่น 92
รูปที่ 3.95 สั่งโปรแกรมเพื่อทำการดาวน์โหลดไฟล์ apk 92
รูปที่ 3.96 Path ของไฟล์ apk 93
รูปที่ 3.97 แอพพลิเคชั่นในโทรศัพท์มือถือระบบปฏิบัติการณ์ Android 93
รูปที่ 3.98 เลือกระบบปฏิบัติการณ์ 94
รูปที่ 3.99 ชื่อ Android package name ใน Flutter 94
รูปที่ 3.100 ตั้งชื่อแอพพลิเคชั่นใน Firebase 95
รูปที่ 3.101 ดาวน์โหลดไฟล์ .json 95
รูปที่ 3.102 นำไฟล์ไปไว้ใน app/src 96
รูปที่ 3.103 เพิ่ม Path ใน android/app/build.gradle 96
รูปที่ 3.104 เพิ่ม Path ใน build.gradle 97
รูปที่ 3.105 กดที่ Continue to console 97

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.106 เพิ่มตัว Extension สำหรับ Firebase 98
รูปที่ 3.107 รูปภาพแสดงการนำอุปกรณ์ไปประกอบในกล่องคอนโทล 98
รูปที่ 3.108 ท่อ PVC ที่ใช้ประกอบโครงของตู้ปลา 99
รูปที่ 3.109 รูปภาพเมื่อประกอบอุปกรณ์เสร็จแล้ว 99
รูปที่ 4.1 ภาพของน้ำที่ใช้ทำการทดลอง 100
รูปที่ 4.2 ภาพขณะวัดอุณหภูมิของน้ำอุณหภูมิปกติด้วยเทอร์โมมิเตอร์ 101
รูปที่ 4.3 ภาพขณะวัดอุณหภูมิของน้ำเย็นด้วยเทอร์โมมิเตอร์ 102
รูปที่ 4.4 ภาพขณะวัดอุณหภูมิของน้ำอุณหภูมิปกติด้วยเซนเซอร์ 103
รูปที่ 4.5 ภาพขณะวัดอุณหภูมิของน้ำเย็นด้วยเซนเซอร์ 103
รูปที่ 4.6 ภาพของน้ำที่มี pH ค่าต่างๆ 104
รูปที่ 4.7 ภาพขณะวัด pH = 6.86 ด้วยเครื่องมือวัด pH 104
รูปที่ 4.8 ภาพขณะวัด pH = 4.01 ด้วยเครื่องมือวัด pH 105
รูปที่ 4.9 ภาพขณะวัด pH = 9.18 ด้วยเครื่องมือวัด pH 106
รูปที่ 4.10 ภาพขณะวัด pH ด้วย pH Sensor 107
รูปที่ 4.11 ภาพของน้ำ 3 ชนิดที่ต่างกัน 108
รูปที่ 4.12 ภาพขณะวัดค่า TDS ของน้ำดื่มด้วย TDS Sensor 108
รูปที่ 4.13 ภาพขณะวัดค่า TDS ของน้ำปะปาด้วย TDS Sensor 109
รูปที่ 4.14 ภาพขณะวัดค่า TDS ของน้ำผสมสบู่ด้วย TDS Sensor 109
รูปที่ 4.15 โค้ดที่ใช้ควบคุมการเปิดหรือปิด Relay ของพัดลมและฮีทเตอร์ 110
รูปที่ 4.16 ภาพของหน้าจอที่ปรับค่าอุณหภูมิไว้ที่ 25 110
รูปที่ 4.17 รูปแสดงค่าได้จากเซ็นเซอร์ 111
รูปที่ 4.18 ภาพขณะที่รีเลย์ตัวที่ควบคุมฮีทเตอร์ทำงาน 111
รูปที่ 4.19 ภาพขณะที่รีเลย์ตัวที่ควบคุมพัดลมทำงาน 112
รูปที่ 4.20 โค้ดที่ใช้ควบคุมการเปิดหรือปิด Relay ของ pH up และ pH down 112
รูปที่ 4.21 ภาพการเทียบค่าเซนเซอร์ที่วัดได้จากจอภาพและ Firebase 113
รูปที่ 4.22 ภาพการเทียบค่าเซนเซอร์ที่วัดได้จากแอพลิเคชั่นและ Firebase 114
รูปที่ 4.23 สั่งผ่านแอพลิเคชั่นให้ Filter และ Light ทำงาน 114
รูปที่ 4.24 สั่งให้ Fan และ Heater ทำงานเพิ่ม 115
รูปที่ 4.25 สั่งให้ pH down และ Filter ทำงาน 115

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่ 1.1 ตารางระยะเวลาการดำเนินงาน..... 3
ตารางที่ 2.1 ตารางแสดงผลของค่า pH ต่างๆที่ปลาดำรงชีวิต.....45

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

1.1 ที่มาและความสำคัญ

ในปัจจุบันเทคโนโลยีได้เข้ามามีบทบาทที่สำคัญกับชีวิตและความเป็นอยู่ของมนุษมากขึ้น ทั้งในด้านการแพทย์ การคมนาคม การสื่อสาร การทำอุตสาหกรรม และความบันเทิงต่างๆ และจะยังคงพัฒนาเทคโนโลยีให้มีความก้าวหน้าให้มากขึ้นมากกว่าปัจจุบันตามกาลเวลา จากเครื่องมือหรืออุปกรณ์ธรรมดาที่ใช้ในชีวิตประจำวันทั่วไปก็ได้พัฒนาให้มีความทันสมัย มีความสะดวกในการใช้งานมากขึ้น เช่น โทรศัพท์มือถือ ในอดีตเป็นเพียงเครื่องมือสื่อสารจากที่ส่งหรือรับสัญญาณเสียงจากที่หนึ่งไปที่หนึ่งเท่านั้น แต่เทคโนโลยีก็ทำให้โทรศัพท์ที่ใช้เพียงติดต่อสื่อสารในอดีต สามารถดู บันทึกภาพหรือวิดีโอ ฟังเพลง เล่นเกมส์ หรือใช้งานเครือข่ายอินเทอร์เน็ตได้ และอื่นๆอีกมากมาย หรืออย่างนาฬิกาข้อมือที่ไม่ใช้แค่ดูเวลาได้เท่านั้น ในปัจจุบันยังสามารถที่จะใช้รับสายโทรศัพท์ เช็คความดัน เช็คอัตราการเต้นของหัวใจได้อีกด้วย ซึ่งนั่นเป็นผลมาจากการที่เทคโนโลยีมีความก้าวหน้าไม่มีหยุด

และเทคโนโลยีที่กำลังเป็นที่กำลังเป็นที่นิยมและเป็นจุดขายของเทคโนโลยีต่างๆในปัจจุบันมากที่สุด ก็คือเทคโนโลนยีที่ใช้อุปกรณ์ในการเชื่อมโยง รับ และส่งข้อมูลโดยใช้เครือข่ายอินเทอร์เน็ต โดยจะนำข้อมูลมาทำการจัดเก็บและประมวลผลแบบออนไลน์ โดยให้ผู้ใช้งานสามารถจัดการ ความคุมข้อมูลได้ตลอดเวลา หรือที่เรียกเทคโนโลยีนี้ว่า อินเทอร์เน็ตของทุกสรรพสิ่ง (Internet Of Things หรือ IOT) ซึ่งเทคโนโลยีนี้ก็ได้ทำให้เกิดการพัฒนาไปเป็นอุปกรณ์ต่างๆที่เรียกว่าเป็นอุปกรณ์อัจฉริยะ เช่น ระบบ Smart Home, Smart farm และ Smart device ต่างๆ ทำเกิดการนำอุปกรณ์เหล่านี้มาใช้ในชีวิตประวันเพื่อความสะดวกสบาย แม่นยำและลดต้นทุนต่างๆไปได้

ซึ่งเทคโนโลยีอัจฉริยะต่างๆที่กล่าวมา ได้นำมาใช้ในด้านต่างๆ เพื่อ ความง่าย และความสะดวกสบายในการจัดการการทำงานต่างๆ ทางผู้จัดทำจึงได้นำเทคโนโลยีข้างต้นมาใช้ในการออกแบบระบบการเลี้ยงปลาและสัตว์น้ำ รวมถึงพืชน้ำ โดยเห็นว่าปลาเป็นสัตว์เลี้ยงชนิดหนึ่งของมนุษย์ โดยสามารถให้ความบันเทิง ให้ความความสวยงามแก่สถาณที่ รวมไปถึงการเลี้ยงเพื่อเป็นอาหารหรือในด้านการเกษตรกรรม โดยสภาพแวดล้อมหรือที่อยู่อาศัยของปลาที่เป็นสัตว์น้ำ จำเป็นจะต้องมีน้ำซึ่งเป็นสิ่งที่สำคัญในการเป็นที่อยู่อาศัยของสัตว์น้ำอย่างปลาต่างๆ และสภาพแวดล้อมที่ปลาชนิดนั้นๆต้องการ ดังนั้น จึงเป็นที่มาของโครงงานระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ควบคุมตู้ปลาอัจฉริยะ โดยใช้ระบบของไมโครคอนโทลเลอร์และเทคโนโลยี IOT ในการควบคุม เพื่อให้การเลี้ยงปลาสะดวกสบาย และสามารถควบคุมสภาพแวดล้อมในตู้ปลาให้เป็นไปอย่างมีประสิทธิมากขึ้น

1.2 วัตถุประสงค์

1. ศึกษาการทำงานของเซนเซอร์แบบต่างๆ
2. ฝึกทักษะการเขียนโปรแกรมเพื่อควบคุมไมโครคอนโทลเลอร์
3. ศึกษาการต่อวงจรอิเล็กทรอนิกส์ เพื่อให้ทำงานตามที่ต้องการ
4. ศึกษาการทำงานของระบบ iot

1.3 ขอบเขตการศึกษา

1.แสดงค่าจากเซนเซอร์ต่างๆผ่านจอ และส่งค่าผ่านทางเครือข่ายอินเตอร์เน็ต ได้อย่างถูกต้อง

2.ใช้ไมโครคอนโทลเลอร์ควบคุมต่างๆในการรักษาสภาพแวดล้อมของน้ำที่ปลาในตู้ต้องการ

3.สามารถควบคุมการทำงานของอุปกรณ์ต่างๆในระยะไกลผ่านระบบเครือข่ายอินเตอร์เน็ต

1.4 ประโยชน์ที่คาดว่าจะได้รับ

1. มีทักษะด้านการเขียนโปรแกรมมากขึ้น
2. มีความรู้เรื่องเซนเซอร์เพื่อไปใช้ในงานอื่นๆได้
3. สามารถแก้ปัญหาต่างๆในด้านอิเล็กทรอนิกส์ได้
4. ตัวชิ้นงานสามารถใช้งานได้จริง และสามารถนำไปต่อยอดเพื่อสร้างรายได้ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.5 ระยะเวลาในการทำโครงงาน

1 สิงหาคม 2565 – 19 พฤษภาคม 2566

ขั้นตอนการดำเนินงาน	สัปดาห์ที่														
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1. ศึกษาทฤษฎี	←	—	—	→											
2. จัดเตรียมอุปกรณ์					←	→									
3. ปฏิบัติงาน							←	—	—	—	—	→			
4. ทดลองการทำงาน													←	→	
5. แก้ไข สรุปผล													←	—	→
6. ทำรายงานฉบับสมบูรณ์	←	—	—	—	—	—	—	—	—	—	—	—	—	—	→

ตารางที่ 1.1 ตารางระยะเวลาการดำเนินงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

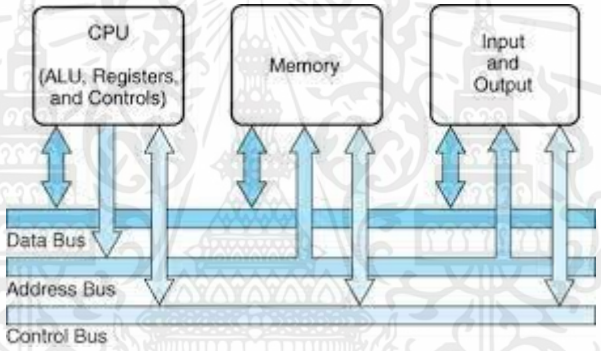
ทฤษฎีอุปกรณ์

2.1 ไมโครคอนโทรลเลอร์

ไมโครคอนโทรลเลอร์ (microcontroller, μC, MCU) คือ ชื่อเรียกของคอมพิวเตอร์ขนาดเล็ก ภายในตัวไมโครคอนโทรลเลอร์จะประกอบไปด้วยหน่วยประมวลผลกลาง (Central Processing Unit, CPU), หน่วยความจำ (Memory), พอร์ต (Port), บัส (BUS) และวงจรกำเนิดสัญญาณนาฬิกา (Clock) บรรจุรวมเข้าไว้ในตัวถังเดียวกัน โดยอุปกรณ์ทั้งห้าส่วนนี้เป็นส่วนประกอบที่สำคัญของระบบคอมพิวเตอร์

2.1.1 โครงสร้างโดยทั่วไปของไมโครคอนโทรลเลอร์

โครงสร้างโดยทั่วไปของไมโครคอนโทรลเลอร์สามารถแบ่งออกได้เป็น 5 ส่วน ดังนี้



รูปที่ 2.1 โครงสร้างโดยทั่วไปของไมโครคอนโทรลเลอร์

(ที่มา : 280661.pdf (rtc.ac.th))

2.1.1.1 หน่วยประมวลผลกลาง (CPU)

หน่วยประมวลผลกลาง คือ ส่วนที่ทำหน้าที่ปฏิบัติตามคำสั่งที่รับมาจากหน่วยรับข้อมูลและควบคุมการปฏิบัติงานของคอมพิวเตอร์ ฮาร์ดแวร์ที่สำคัญในหน่วยประมวลผลกลาง คือ ไมโครโพรเซสเซอร์ หน่วยประมวลผลกลางมีส่วนประกอบที่สำคัญ 2 ส่วน คือ

1.หน่วยควบคุม (Control Unit) ทำหน้าที่ควบคุมการทำงานของอุปกรณ์ต่างๆในระบบทั้งหมดให้ทำงานอย่างถูกต้อง

2.หน่วยคำนวณ (Arithmetic Logic Unit) ทำหน้าที่ประมวลผลข้อมูลทางคณิตศาสตร์และตรรกะ เช่น การดำเนินการทางคณิตศาสตร์ ได้แก่ การบวก ลบ คูณ หาร, การกระทำทางตรรกะ เช่น AND OR, การ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เปรียบเทียบ เช่น การเปรียบเทียบค่าของข้อมูล 2 ตัวว่ามีค่าเท่ากัน มากกว่า หรือน้อยกว่า ไม่ว่าข้อมูลจะเป็นตัวเลข หรือตัวอักษรก้สามารถเปรียบเทียบได้, การเลื่อนข้อมูล (Shift), การเพิ่มและการลด (Increment and Decrement), การตรวจสอบบิท (Test Bit) เป็นต้น

หน่วยประมวลผลกลางจะทำงานเป็น 4 ขั้นตอน โดยขั้นตอนที่ 1 และ 2 จะใช้หน่วยควบคุมในการดำเนินงาน ส่วนขั้นตอนที่ 3 และ 4 จะใช้หน่วยคำนวณในการดำเนินงาน ดังนี้

ขั้นตอนที่ 1 หน่วยควบคุมเข้าถึงข้อมูลและคัดแยกคำสั่งจากหน่วยความจำ

ขั้นตอนที่ 2 คำสั่งถูกตีความ เพื่อให้คอมพิวเตอร์รู้ว่าจะต้องทำงานอะไร แล้วเลือกข้อมูลที่ต้องใช้ในการประมวลผล แล้วกำหนดตำแหน่งของคำสั่งถัดไป

ขั้นตอนที่ 3 ปฏิบัติตามคำสั่งที่ตีความได้ ทั้งการคำนวณทางคณิตศาสตร์และการเปรียบเทียบ

ขั้นตอนที่ 4 เก็บผลลัพธ์ที่ประมวลผลได้ไว้ในหน่วยความจำหลัก

2.1.1.2 หน่วยความจำ (Memory)

หน่วยความจำ (Memory) เป็นที่เก็บโปรแกรมข้อมูลและผลลัพธ์ไว้ในคอมพิวเตอร์ หน่วยความจำแบ่งออกเป็น 2 ประเภท คือ

1. หน่วยความจำหลัก จะอยู่ภายในตัวเครื่อง แบ่งออกเป็น

- ROM (Read Only memory) คือ หน่วยความจำที่จะถูกอ่านได้อย่างเดียวเท่านั้นโดยจะเก็บคำสั่งหรือโปรแกรมไว้อย่างถาวร แม้ปิดเครื่องก็จะไม่ถูกลบ

- RAM (Random access memory) คือ หน่วยความจำที่ใช้ในการจดจำข้อมูลหรือคำสั่ง ขณะที่เครื่องทำงานซึ่งสามารถเปลี่ยนแปลงข้อมูลหรือคำสั่งได้ตลอดเวลาที่ยังเปิดเครื่องแต่เมื่อปิดเครื่องคอมพิวเตอร์ข้อมูลและโปรแกรมจะถูกลบหายไป

2. หน่วยความจำรอง (Secondary Storage) เปรียบเสมือนสมุดบันทึกสำหรับเก็บโปรแกรมและข้อมูลไว้ใช้ในโอกาสต่อไป

2.1.1.3 พอร์ต (Port)

มี 2 แบบ คือ พอร์ตอินพุต (Input Port) และพอร์ตเอาต์พุต (Output Port) พอร์ตจะใช้ในการเชื่อมต่อกับอุปกรณ์ภายนอก

2.1.1.4 บัส (BUS)

บัส (BUS) คือ เส้นทางการแลกเปลี่ยนสัญญาณข้อมูลระหว่าง ซีพียู หน่วยความจำและพอร์ต เป็นลักษณะของสายสัญญาณจำนวนมากอยู่ภายในตัวไมโครคอนโทรลเลอร์ บัสแบ่งเป็นบัสข้อมูล (Data Bus), บัส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แอดเดรส (Address Bus) และบัสควบคุม (Control Bus) ขนาดของบัสจะขึ้นอยู่กับความสามารถการประมวลผลของซีพียู

2.1.1.5 วงจรกำเนิดสัญญาณนาฬิกา (Clock)

เป็นองค์ประกอบที่สำคัญมากอีกส่วนหนึ่ง เนื่องจากการทำงานที่เกิดขึ้นในตัวไมโครคอนโทรลเลอร์จะขึ้นอยู่กับการกำหนดจังหวะ หากสัญญาณนาฬิกามีความถี่สูง จังหวะการทำงานก็จะสามารถทำได้ถี่ขึ้น ส่งผลให้ไมโครคอนโทรลเลอร์ตัวนั้น มีความเร็วในการประมวลผลสูงตามไปด้วย

2.1.2 สถาปัตยกรรมของไมโครคอนโทรลเลอร์

สถาปัตยกรรมของไมโครคอนโทรลเลอร์ คือ รูปแบบการออกแบบพื้นฐานของคอมพิวเตอร์ที่ใช้หน่วยประมวลผลกับพื้นที่บันทึกความจำเป็นโครงสร้างในการเก็บคำสั่งและข้อมูล สถาปัตยกรรมของไมโครคอนโทรลเลอร์แบ่งออกเป็น 2 แบบหลักๆ คือ

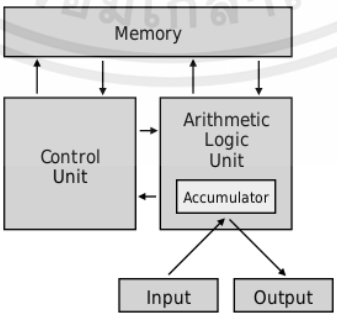
2.1.2.1 สถาปัตยกรรมฟอนนอยมันน์

โครงสร้างของสถาปัตยกรรมฟอนนอยมันน์ ประกอบด้วยรูปแบบการทำงานหลักๆ 3 แบบ ได้แก่

1.ตัวเครื่องคอมพิวเตอร์ ซึ่งประกอบด้วย 4 ส่วนย่อย คือ หน่วยความจำ (Memory), หน่วยคำนวณและตรรกะ (Arithmetic Logic Unit หรือ ALU), หน่วยควบคุม (Control unit), ระบบรับค่าและแสดงผล (Input/Output System หรือ I/O)

2.โปรแกรม ซึ่งจะถูกเก็บไว้ในหน่วยความจำ จนกว่าจะถูกเรียกใช้

3.คำสั่งของโปรแกรม ซึ่งจะทำงานตามลำดับขั้นตอน เมื่อโปรแกรมถูกเรียกใช้



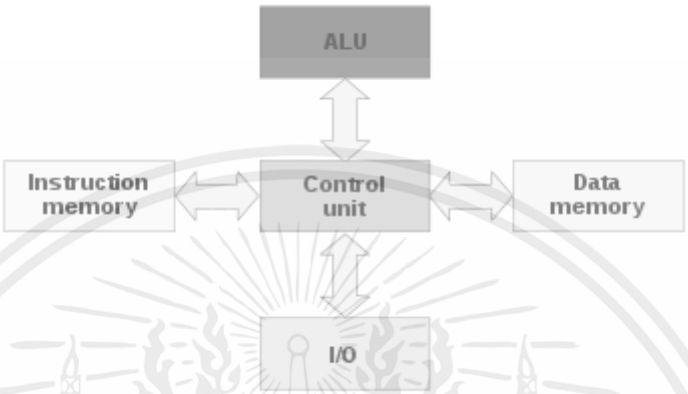
รูปที่ 2.2 สถาปัตยกรรมฟอนนอยมันน์

(ที่มา : สถาปัตยกรรมฟอนนอยมันน์ - วิกิพีเดีย (wikipedia.org),2558)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.2.2 สถาปัตยกรรมฮาร์วาร์ด

สถาปัตยกรรมฮาร์วาร์ดเป็นสถาปัตยกรรมคอมพิวเตอร์ที่แยกการจัดเก็บข้อมูลและสัญญาณทางเดินสำหรับคำแนะนำและข้อมูล ซึ่งแตกต่างกับสถาปัตยกรรมฟอนนอยมันน์ที่คำสั่งโปรแกรมและข้อมูลจะใช้หน่วยความจำและเส้นทางเดียวกัน



รูปที่ 2.3 สถาปัตยกรรมฮาร์วาร์ด

(ที่มา : สถาปัตยกรรมฮาร์วาร์ด รายละเอียดหน่วยความจำและความเร็ว (hmong.in.th),1988)

2.1.3 ไมโครโปรเซสเซอร์

ไมโครโปรเซสเซอร์ คือ ตัวชิปเปล่าๆหรือตัวชิปที่อยู่บนไมโครคอนโทรลเลอร์ อาจกล่าวได้ว่าไมโครคอนโทรลเลอร์คือการเพิ่มขาของไมโครโปรเซสเซอร์ขึ้นมาเพื่อให้ใช้งานได้ง่ายยิ่งขึ้น เพราะถ้าเป็นไมโครเซสเซอร์ตัวเปล่าๆนั้น จะต้องต่อขาอุปกรณ์โดยใช้ลายทองแดงหรือ PCB รวมถึงการเบิร์นโค้ดด้วย ซึ่งจะยุ่งยากกว่าการใช้ไมโครคอนโทรลเลอร์ที่สามารถต่อขาอุปกรณ์โดยใช้สายจั๊มเปอร์จากภายนอก และสามารถเบิร์นโค้ดจากคอมพิวเตอร์โดยใช้สาย USB ชนิดต่างๆได้โดยตรง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 ESP32

ESP32 เป็นชื่อของไอซีไมโครคอนโทรลเลอร์ที่รองรับการเชื่อมต่อ Wi-Fi และ Bluetooth 4.2 BLE ภายในตัว ผลิตโดยบริษัท Espressif จากประเทศจีน โดยตัวไอซี ESP32 มีสเปคโดยละเอียด ดังนี้

-ซีพียูใช้สถาปัตยกรรม Tensilica LX6 แบบ 2 แกนสมอง สัญญาณนาฬิกา 240MHz

-มีแรมในตัว 512KB

-รองรับการเชื่อมต่อรอมภายนอกสูงสุด 16MB

-รองรับการใช้งานทั้งในโหมด Station softAP และ Wi-Fi direct

-มีบลูทูธในตัว รองรับการใช้งานในโหมด 2.0 และโหมด 4.0 BLE

-ใช้แรงดันไฟฟ้าในการทำงาน 2.6V ถึง 3V

-ทำงานได้ที่อุณหภูมิ -40°C ถึง 125°C

-สามารถใช้ซอฟแวร์ Arduino IDE ในการอัพโค้ดได้

รูปที่ 2.4 ชิป ESP32

(ที่มา : ESP32 เบื้องต้น :: บทที่ 1 แนะนำ ESP32 - ArtronShop บอร์ดอิเล็กทรอนิกส์ Arduino ESP32 ESP8266 : Inspired by LnwShop.com,2564)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.1 ชุดซอฟแวร์พัฒนา ESP32

-LuaNode

เป็นชื่อของชุดพัฒนา ESP32 ที่นำ Runtime ของภาษา Lua มาลงใน ESP32 ทำให้ ESP32 ใช้ภาษา Lua ได้ พัฒนาโดยบริษัท DOIT ที่ทำบอร์ดพัฒนา ESP32 ในชื่อ DOIT ESP32 Development Board โดยความสามารถของ LuaNode คือรองรับคำสั่งที่ใช้บน Lua จริงๆเกือบทุกคำสั่ง และรองรับการควบคุม Wi-Fi เต็มรูปแบบ

รูปที่ 2.5 DOIT ESP32 Development Board

(ที่มา : ESP32 เบื้องต้น :: บทที่ 1 แนะนำ ESP32 - ArtronShop บอร์ดอิเล็กทรอนิกส์ Arduino ESP32 ESP8266 : Inspired by LnwShop.com,2564)

-MicroPython-ESP32

ชุดพัฒนาที่พยายามสร้างตัว Runtime ของภาษา Python 3 บนตัวไมโครคอนโทรลเลอร์ต่างๆ พัฒนาโดย MicroPython MicroPython-ESP32 รองรับการใช้งานพื้นฐานภาษา Python ส่วนใหญ่ได้ รองรับการจัดการ Wi-Fi การเชื่อมต่อ I2C SPI ADC และการควบคุม GPIO เต็มรูปแบบ

-Espruino on ESP32

เป็นชุดพัฒนาที่พยายามทำให้สามารถใช้ภาษา JavaScript ในการสั่งงานได้ โดยโครงการ Espruino ได้ทำตัว Runtime ขึ้นมาใช้กับ ESP32 และไมโครคอนโทรลเลอร์ตัวอื่นๆ โครงการนอกจากจะพัฒนาตัวเฟิร์มแวร์ Runtime แล้ว ยังได้พัฒนา Espruino Web IDE ซึ่งเป็นโปรแกรม IDE แบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เว็บแอพพลิเคชั่นที่ใช้งานร่วมกับ Google Chrome ด้วย ในการติดตั้ง จะต้องติดตั้งผ่าน Chrome เว็บสโตร์ รองรับการเขียนโปรแกรมแบบ Text และภาษาบล็อก (Block)

-Arduino IDE

Arduino IDE เป็นโปรแกรมที่ใช้งานในลักษณะ Open source ซึ่ง Arduino IDE จะทำหน้าที่ติดต่อระหว่างคอมพิวเตอร์ ไม่ว่าจะเป็นระบบ Windows, Mac OS X หรือ Linux กับ บอร์ด Arduino ซึ่งโปรแกรมนี้ออกแบบให้ง่ายต่อการเขียนโค้ดและอัปโหลดโปรแกรมที่เราเขียนเข้าสู่บอร์ด Arduino

2.2.2 โมดุลของ ESP32

2.2.2.1 ESP-WROOM-32

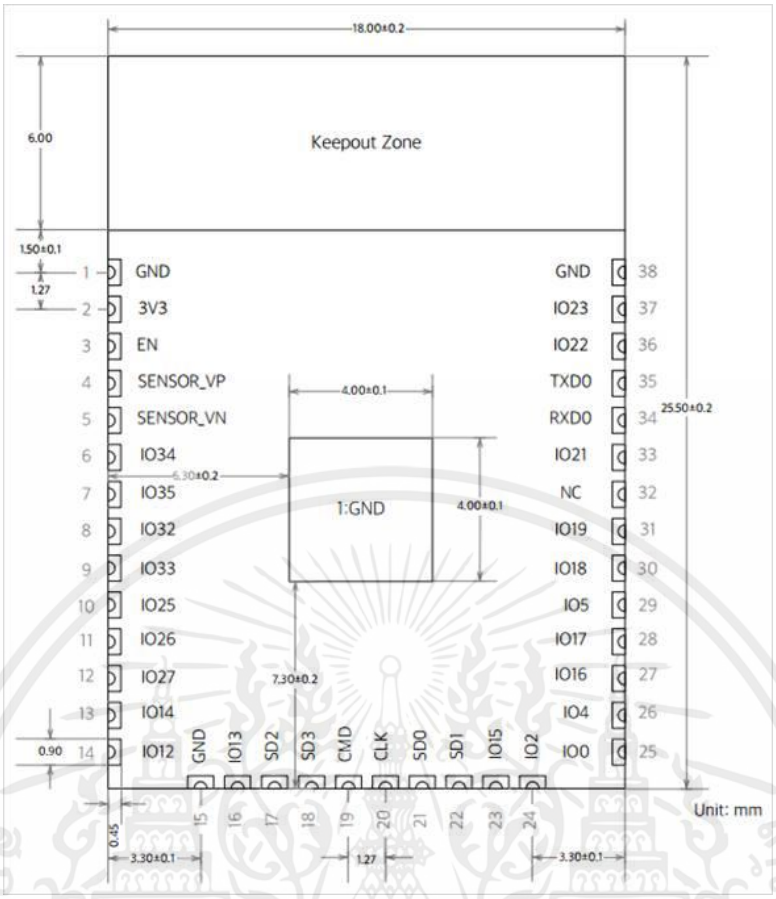
โมดูล ESP32 รุ่นนี้ ผลิตโดยบริษัท Espressif ที่เป็นผู้ผลิตชิปไอซี ESP32 มีอยู่ด้วยกัน 2 เวอร์ชั่น

รูปที่ 2.6 โมดูล ESP-WROOM-32

(ที่มา : ESP32 เบื้องต้น :: บทที่ 2 บอร์ดพัฒนา ESP32 - ArtronShop บอร์ดอิเล็กทรอนิกส์ Arduino ESP32 ESP8266 : Inspired by LnwShop.com,2564)

โครงสร้างโดยละเอียดของ ESP-WROOM-32 มีดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

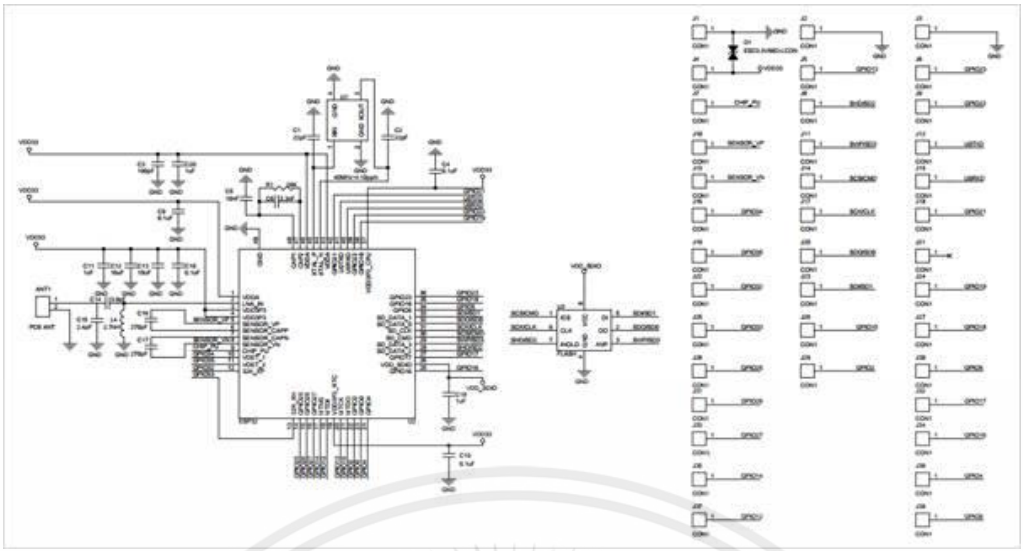


รูปที่ 2.7 โครงสร้างของ ESP-WROOM-32

(ที่มา : ESP32 เบื้องต้น :: บทที่ 2 บอร์ดพัฒนา ESP32 - ArtronShop บอร์ดอิเล็กทรอนิกส์ Arduino ESP32 ESP8266 : Inspired by LnwShop.com,2564)

ตัวโมดูลมาพร้อมกับเสาอากาศแบบ PCB บนตัว มีกรอบคุมอุปกรณ์ทั้งหมดบนโมดูล ทำให้สัญญาณรบกวนน้อยลง และทำให้การทำงานโดยรวมมีเสถียรภาพมากยิ่งขึ้น ESP-WROOM-32 ได้รับความนิยมอย่างมากในการใช้งาน เนื่องจากสามารถผลิตได้ทันความต้องการ และมีราคาถูกกว่าโมดูลที่ผลิตโดยผู้ผลิตอื่นๆ นอกจากนี้ ตำแหน่งขาต่าง ๆ ของ ESP-WROOM-32 ยังถือเป็นมาตรฐานที่โมดูลจากผู้ผลิตอื่นทำตามอีกด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.8 วงจรของ ESP-WROOM-32

(ที่มา : ESP32 **เบื้องต้น :: บทที่ 2 บอร์ดพัฒนา** ESP32 - ArtronShop **บอร์ดอิเล็กทรอนิกส์** Arduino ESP32 ESP8266 : Inspired by LnwShop.com,2564)

2.2.2.2 ESP32S

รูปที่ 2.9 หน้าตาของโมดูล ESP32S

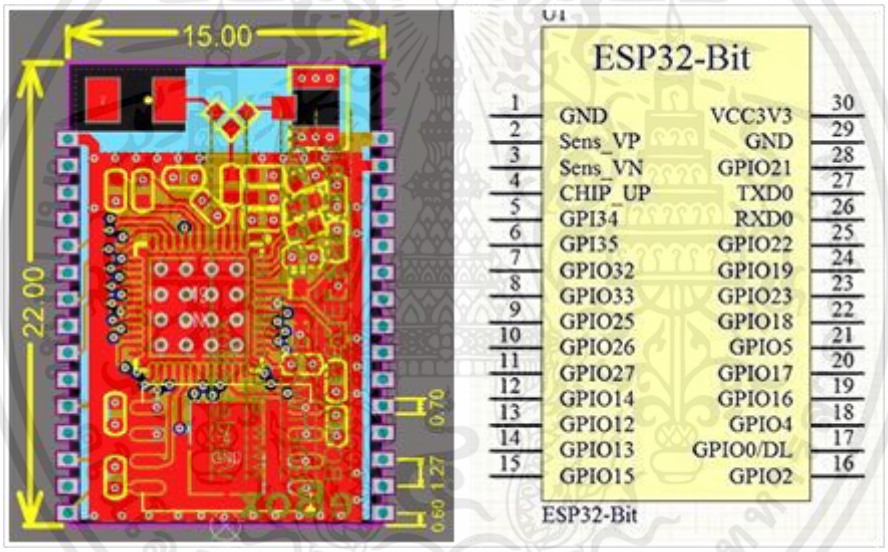
(ที่มา : ESP32 **เบื้องต้น :: บทที่ 2 บอร์ดพัฒนา** ESP32 - ArtronShop **บอร์ดอิเล็กทรอนิกส์** Arduino ESP32 ESP8266 : Inspired by LnwShop.com,2564)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เป็นโมดูล ESP32 ที่ผลิตโดย Ai-Thinker ขาใช้งาน รวมถึงขนาดของตัวถัง ลายวงจรต่าง ๆ เหมือนกับ ESP-WROOM-32 ทุกประการ ทำให้สามารถนำไปใช้แทนกันได้เลย แต่ ESP32S มีราคาที่แพงกว่า ESP-WROOM-32 ทำให้ได้รับความนิยมในการนำไปใช้บนโมดูลน้อยกว่า ESP-WROOM-32

2.2.2.3 ESP32-Bit

ด้วยความร่วมมือของ eBox และ Widora ผู้ผลิตบอร์ดไมโครคอมพิวเตอร์ในจีน ทำให้โมดูล ESP32-Bit เกิดขึ้นมา ข้อดีของโมดูล ESP32-Bit นี้ คือมีขนาดเล็กกว่า ESP-WROOM-32 ทำให้เมื่อนำไปออกแบบเป็นบอร์ดพัฒนา จะทำให้บอร์ดมีขนาดที่เล็กลงได้ แต่ข้อเสียคือมีการตัดขาใช้งานบางส่วนออกไป จึงอาจจะทำให้ไม่สามารถ ESP32 ได้เต็มที่มากนัก ข้อดีอีกข้อของโมดูล ESP32-Bit คือ สามารถเลือกได้ว่าจะใช้เสาเพิ่มโดยใช้คอนเนคเตอร์แบบ IPEX หรือเลือกใช้เสาอากาศแบบเซรามิคบนตัวโมดูล ซึ่ง ESP-WROOM-32 และ ESP32S ไม่สามารถทำได้

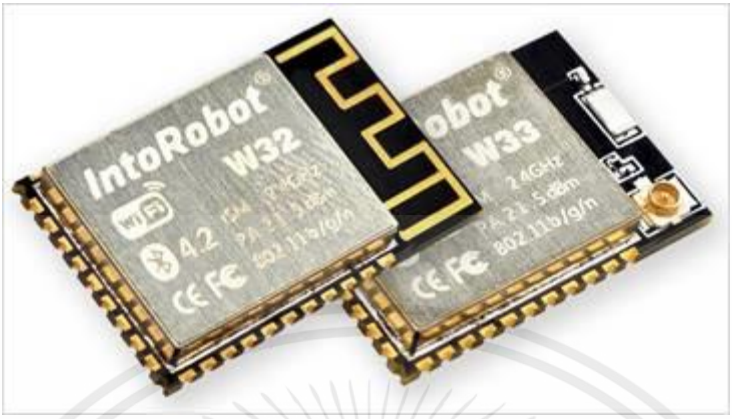


รูปที่ 2.10 โครงสร้างของ ESP32-Bit (รูปซ้าย) และตำแหน่งขาต่อใช้งาน (รูปขวา)

(ที่มา : ESP32 เบื้องต้น :: บทที่ 2 บอร์ดพัฒนา ESP32 - ArtronShop บอร์ดอิเล็กทรอนิกส์ Arduino ESP32 ESP8266 : Inspired by LnwShop.com,2564)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

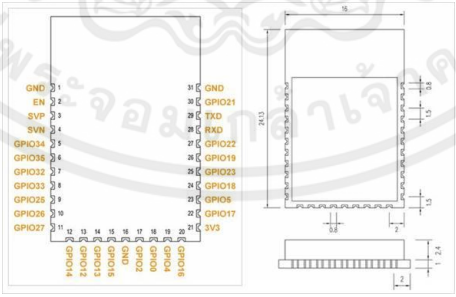
2.2.2.4 IntoRobot-W32/W33



รูปที่ 2.11 โมดูลรุ่น IntoRobot-W32 และรุ่น IntoRobot-W33

(ที่มา : ESP32 **เบื้องต้น :: บทที่ 2 บอร์ดพัฒนา** ESP32 - ArtronShop **บอร์ดอิเล็กทรอนิกส์** Arduino ESP32 ESP8266 : Inspired by LnwShop.com,2564)

เป็นโมดูล ESP32 ที่ผลิตโดย IntoRobot มีอยู่ 2 รุ่นคู่กัน คือรุ่น W32 และ W33 ข้อแตกต่างของทั้ง 2 รุ่น คือรุ่น W32 จะใช้เสาอากาศแบบ PCB ส่วนรุ่น W33 จะมีเสาแบบเซรามิคเป็นเสาหลักในการรับสัญญาณ และสามารถเปลี่ยนไปใช้คอนเนคเตอร์แบบ IPEX สำหรับต่อเสาเพิ่มเติมภายนอกได้ ตำแหน่งขาและขนาดจะมีความแตกต่างจากโมดูลทุกรุ่นที่กล่าวมา คือมีขาต่อใช้งานทั้งหมด 31 ขา น้อยกว่า ESP-WROOM-32 กับ ESP32S แต่มากกว่า ESP32-Bit และมีความกว้าง 16 มิลิเมตร เล็กกว่า ESP-WROOM-32 กับ ESP32S แต่ใหญ่กว่า ESP32-Bit



รูปที่ 2.12 ตำแหน่งขาต่าง ๆ และโครงสร้างของ IntoRobot-W32/W33

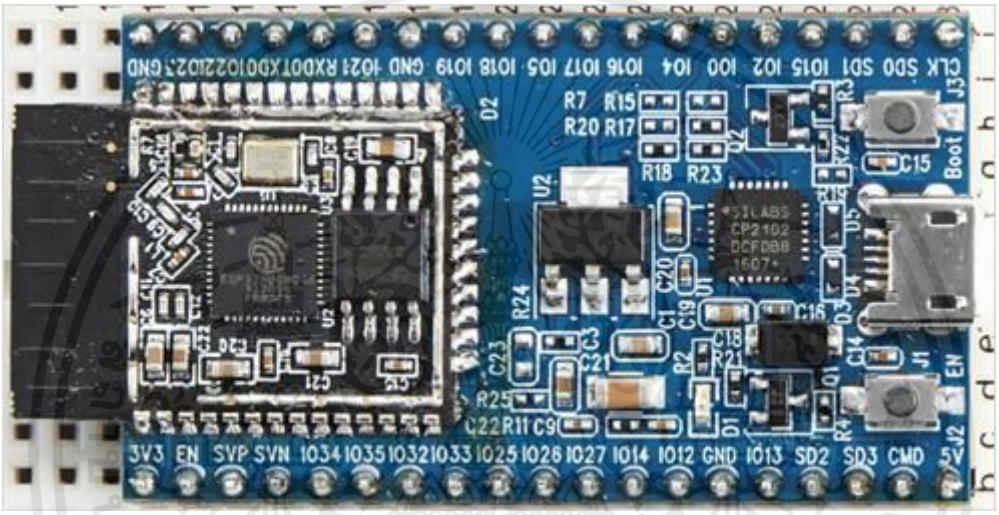
(ที่มา : ESP32 **เบื้องต้น :: บทที่ 2 บอร์ดพัฒนา** ESP32 - ArtronShop **บอร์ดอิเล็กทรอนิกส์** Arduino ESP32 ESP8266 : Inspired by LnwShop.com,2564)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.3 บอร์ดพัฒนาของ ESP32

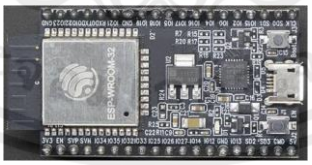
หลังจากชิปไอซี ESP32 เปิดตัวได้ไม่นาน ก็มีผู้ผลิตหลายรายที่ให้การตอบรับโดยการผลิตบอร์ดพัฒนา ESP32 ออกมา ช่วยให้ ESP32 สามารถนำมาพัฒนาได้ง่ายมากขึ้น ในแต่ละบอร์ดก็จะมีความแตกต่างกันในเรื่องของฟิเจอร์เพิ่มเติม เช่น เซ็นเซอร์วัดอุณหภูมิที่เพิ่มขึ้นมา มีคลิสตอล 32.768kHz สำหรับจ่ายสัญญาณนาฬิกาให้วงจร RTC ภายใน ESP32 เรื่องขนาด และรุ่นของโมดูล ESP32 ที่เลือกใช้ สำหรับบอร์ดพัฒนา ESP32 ที่จะยกตัวอย่างนี้เป็นบอร์ดที่เป็นที่นิยมใช้งาน

2.2.3.1 ESP32-DevKitC



รูปที่ 2.13 บอร์ด ESP32-DevKitC เวอร์ชั่นแรก

(ที่มา : ESP32 เบื้องต้น :: บทที่ 2 บอร์ดพัฒนา ESP32 - ArtronShop บอร์ดอิเล็กทรอนิกส์ Arduino ESP32 ESP8266 : Inspired by LnwShop.com,2564)



รูปที่ 2.14 บอร์ด ESP32-DevKitC เวอร์ชั่นที่สอง เป็นเวอร์ชั่นที่จำหน่ายในปัจจุบัน

(ที่มา : ESP32 เบื้องต้น :: บทที่ 2 บอร์ดพัฒนา ESP32 - ArtronShop บอร์ดอิเล็กทรอนิกส์ Arduino ESP32 ESP8266 : Inspired by LnwShop.com,2564)

บอร์ดพัฒนา ESP32 ที่ผลิตโดยบริษัท Espressif ที่เป็นผู้ผลิตชิปไอซี ESP32 เป็นบอร์ดพัฒนา ESP32 บอร์ดแรก ทำให้ตำแหน่งขาต่าง ๆ ของ ESP32-DevKitC ถูกใช้เป็นมาตรฐาน บอร์ดพัฒนา ESP32 ส่วนใหญ่

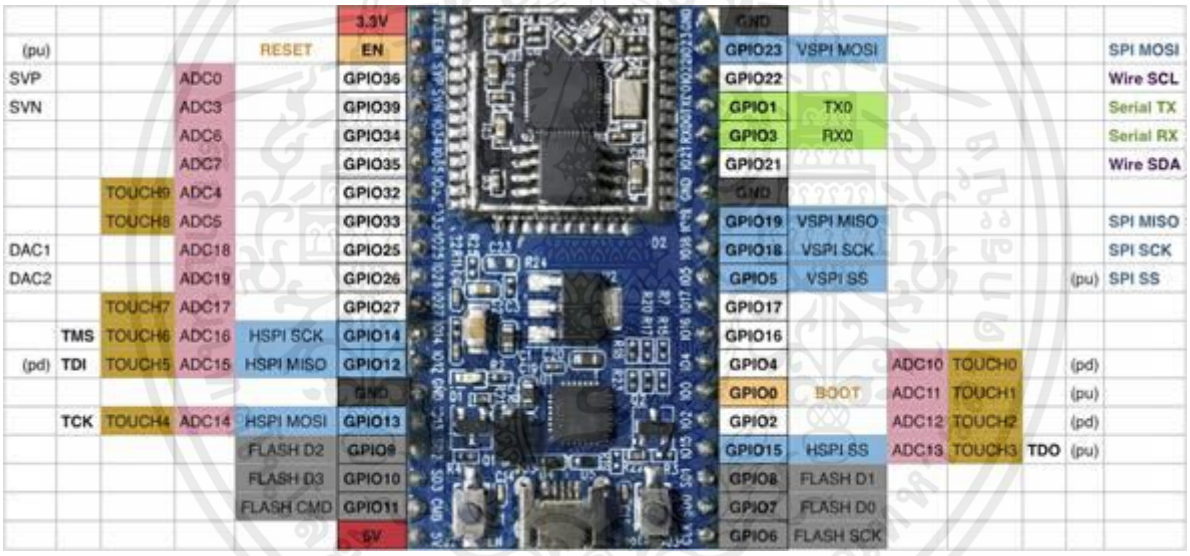
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะมีขนาด และมีตำแหน่งขาเท่าบอร์ด ESP32-DevKitC ดังนั้นหากไม่สามารถหาซื้อบอร์ด ESP32-DevKitC ได้ ก็สามารถนำบอร์ดจากผู้ผลิตอื่น ๆ มาเสียบแทนกันได้เลย บอร์ดพัฒนา ESP32-DevKitC มีอยู่ด้วยกัน 2 รุ่น คือ

V1 - บอร์ดสีฟ้า เป็นรุ่นแรก และมีร้าน Adafruit ร้านเดียวที่ค้าปลีกให้บุคคลทั่วไปซื้อมาลองใช้งานได้

V2 - เปลี่ยนสีของแผ่นวงจรพิมพ์เป็นสีดำ สามารถหาซื้อได้ตามร้านค้าออนไลน์ในหลาย ๆ ประเทศ

บอร์ด ESP32-DevKitC ใช้โมดูลรุ่น ESP-WROOM-32 ที่บริษัท Espressif เป็นผู้ผลิตขึ้นเอง และใช้ชิปไอซีแปลง USB เป็น UART เบอร์ CP2102 จาก Silicon Labs สามารถเข้าโหมดอัพโหลดโปรแกรมแบบอัตโนมัติโดยใช้วงจรแบบ nodemcu มีรอม 4MB (หรือ 32Mbit) ใช้ไอซีเรกกูเลเตอร์แบบ LDO เบอร์ AMS1117-3.3 รองรับแรงดันเข้าสูงสุด 7V และจ่ายกระแสได้สูงสุด 700mA ใช้พลังงานไฟฟ้าและสื่อสารผ่านพอร์ต MicroUSB มีขาต่อใช้งานทั้งหมด 38 ขา เรียงขาแบบเดียวกับบนโมดูล



รูปที่ 2.15 ตำแหน่งขาบนบอร์ด ESP32-DevKitC

(ที่มา : ESP32 เบื้องต้น :: บทที่ 2 บอร์ดพัฒนา ESP32 - ArtronShop บอร์ดอิเล็กทรอนิกส์ Arduino ESP32 ESP8266 : Inspired by LnwShop.com,2564)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.3.2 SparkFun ESP32 Thing

รูปที่ 2.16 บอร์ด SparkFun ESP32 Thing

(ที่มา : ESP32 เบื้องต้น :: บทที่ 2 บอร์ดพัฒนา ESP32 - ArtronShop บอร์ดอิเล็กทรอนิกส์ Arduino ESP32 ESP8266 : Inspired by LnwShop.com,2564)

บอร์ดพัฒนา ESP32 ที่ผลิตโดยบริษัท SparkFun จากสหรัฐอเมริกา ความพิเศษของบอร์ดนี้ คือมีวงจรชาร์จแบตเตอรี่ลิโพบนบอร์ด ทำให้บอร์ดสามารถใช้งานกับแบตเตอรี่ได้โดยตรงผ่านการเชื่อมต่อแบตเตอรี่โดยใช้ JST คอนเนคเตอร์ และมีคลิสตอล 32.768kHz สำหรับจ่ายสัญญาณนาฬิกาให้วงจร RTC ภายในชิปไอซี ESP32

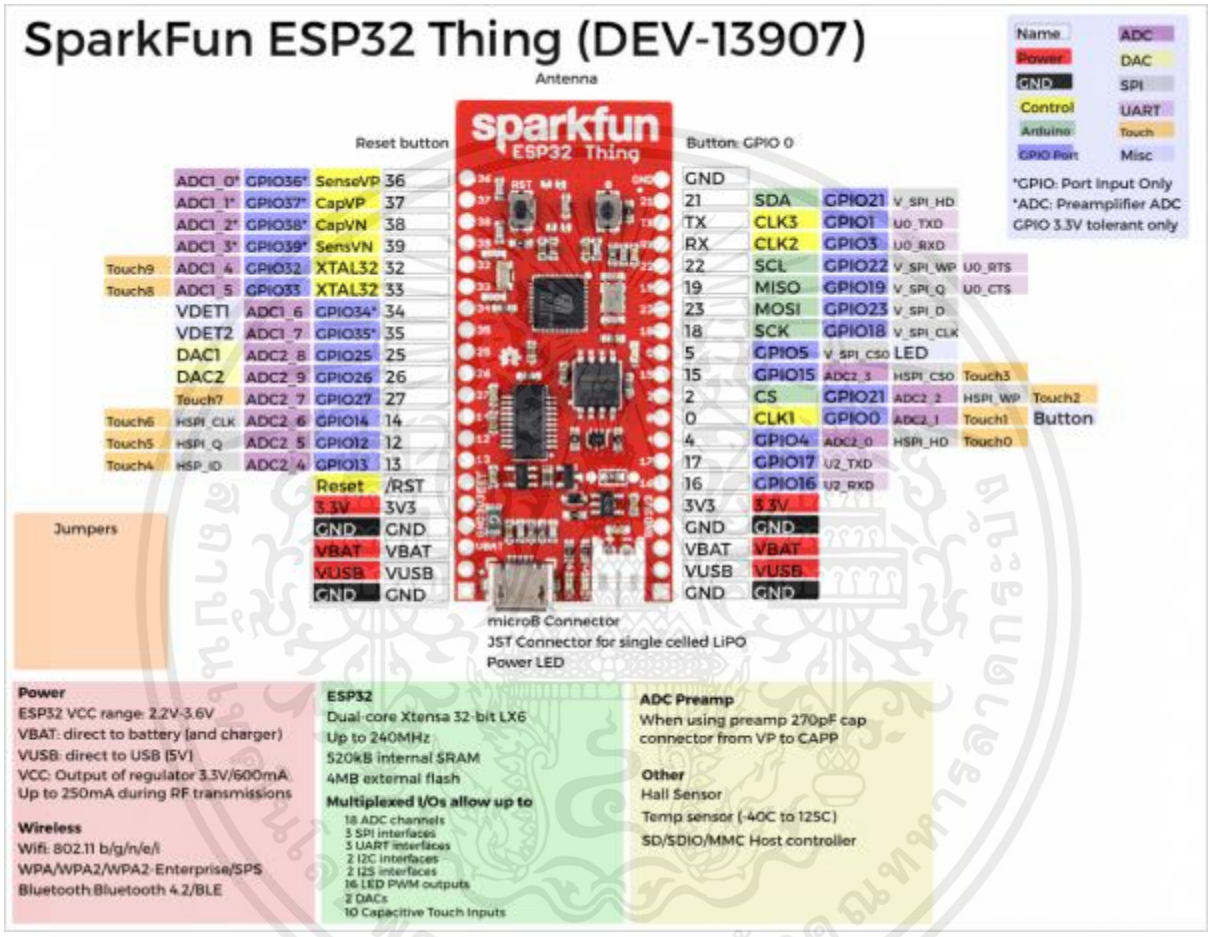
SparkFun นิยมออกแบบบอร์ดพัฒนาโดยการใช้ชิปไอซีโดยตรง ทำให้ตัวบอร์ด SparkFun ESP32 Thing ไม่ใช้โมดูล ESP32 จากผู้ผลิตรายต่างๆ และเลือกใช้การวางชิปไอซี ESP32 ลงบนบอร์ดของตนเอง ข้อดีของการไม่เลือกใช้โมดูลคือสามารถควบคุมขนาดของบอร์ดได้ความต้องการ แต่ก็อาจจะทำให้วงจรทำงานไม่เสถียรเท่าการใช้โมดูลเนื่องจากไม่มีฝาครอบที่จะมากำจัดสัญญาณรบกวนเข้าวงจร

ตัวบอร์ดใช้เสาอากาศแบบ PCB ใช้ชิปไอซีรอมขนาด 4MB ตัวถังแบบ SOIC-8 เชื่อมต่อผ่าน SPI เลือกใช้ชิปไอซีแปลง USB เป็น UART เบอร์ FT231X จากบริษัท FTDI ใช้ชิปไอซีเรกกูเลเตอร์แบบ LDO เบอร์ AP2112K-3.3V ตัวถังแบบ SOT-23 รองรับแรงดันอินพุตสูงสุด 6V และจ่ายกระแสเอาต์พุตได้ไม่เกิน 600mA ส่วนวงจรชาร์จแบตเตอรี่ใช้ชิปไอซีเบอร์ MCP73831 กำหนดกระแสชาร์จไว้ที่ 500mA หลอด LED สีแดง บอกสถานะชิป ESP32 มีไฟเลี้ยง 3.3V และหลอด LED สีเหลือง บอกสถานะการชาร์จแบตเตอรี่ มีคลิ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สตอล 32.768kHz สำหรับจ่ายสัญญาณนาฬิกาให้วงจร RTC ในชิปไอซี ESP32 สามารถเข้าโหมดอัพโหลดโปรแกรมได้อัตโนมัติโดยจัดวงจรแบบ nodemcu ใช้พลังงานไฟฟ้าและสื่อสารผ่านพอร์ต MicroUSB มีขาต่อใช้งานทั้งหมด 40 ขา

บอร์ด SparkFun ESP32 Thing จำหน่ายในเว็บของ SparkFun เอง โดยเปิดให้สั่งซื้อได้เพียงบางประเทศเท่านั้น ในราคา $19.95 ไม่รวมค่าจัดส่ง หรือประมาณ 730 บาท ไม่รวมค่าจัดส่ง



รูปที่ 2.17 Pinout ของบอร์ด SparkFun ESP32 Thing

(ที่มา : ESP32 เบื้องต้น :: บทที่ 2 บอร์ดพัฒนา ESP32 - ArtronShop บอร์ดอิเล็กทรอนิกส์ Arduino ESP32 ESP8266 : Inspired by LnwShop.com,2564)

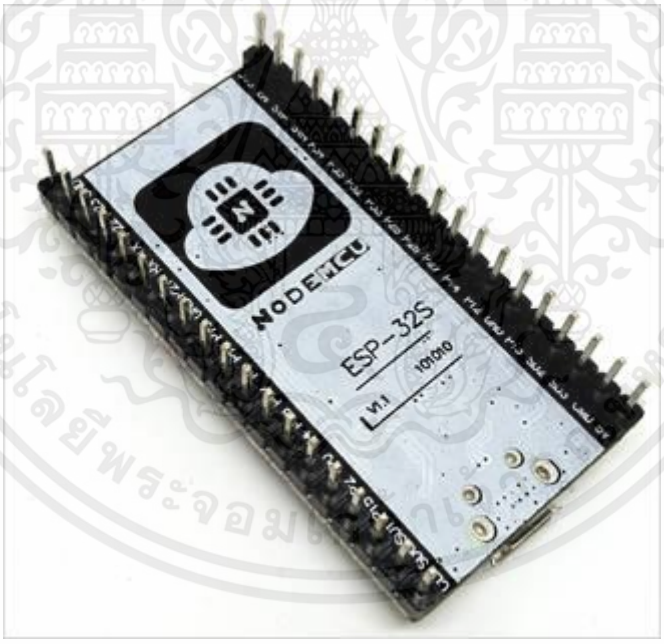
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.3.3 NodeMCU-32S

รูปที่ 2.18 บอร์ด NodeMCU-32S

(ที่มา : ESP32 **เบื้องต้น :: บทที่ 2 บอร์ดพัฒนา** ESP32 - ArtronShop **บอร์ดอิเล็กทรอนิกส์** Arduino ESP32 ESP8266 : Inspired by LnwShop.com,2564)

ผลิตโดย Ai-Thinker ผู้ผลิตโมดูล ESP32S มีตำแหน่งขาที่ตรงกับ ESP32-DevKitC แต่มีความกว้างที่เล็กกว่า ทำให้ไม่สามารถใช้งานแทนกันได้โดยตรง ข้อดีของโมดูลนี้คือมีขนาดเล็ก ไม่มีส่วนของเสาอากาศบนโมดูลยื่นออกมาจากบอร์ดหลัก ใช้เรกกูเลเตอร์ที่รองรับแรงดันไฟฟ้าเข้าได้ถึง 7V แต่มีข้อเสียที่บนบอร์ดไม่มีสกรีนตำแหน่งขา แต่มีสกรีนตำแหน่งขาด้านล่างบอร์ดแทน ทำให้การใช้งานอาจจะไม่สะดวกนัก



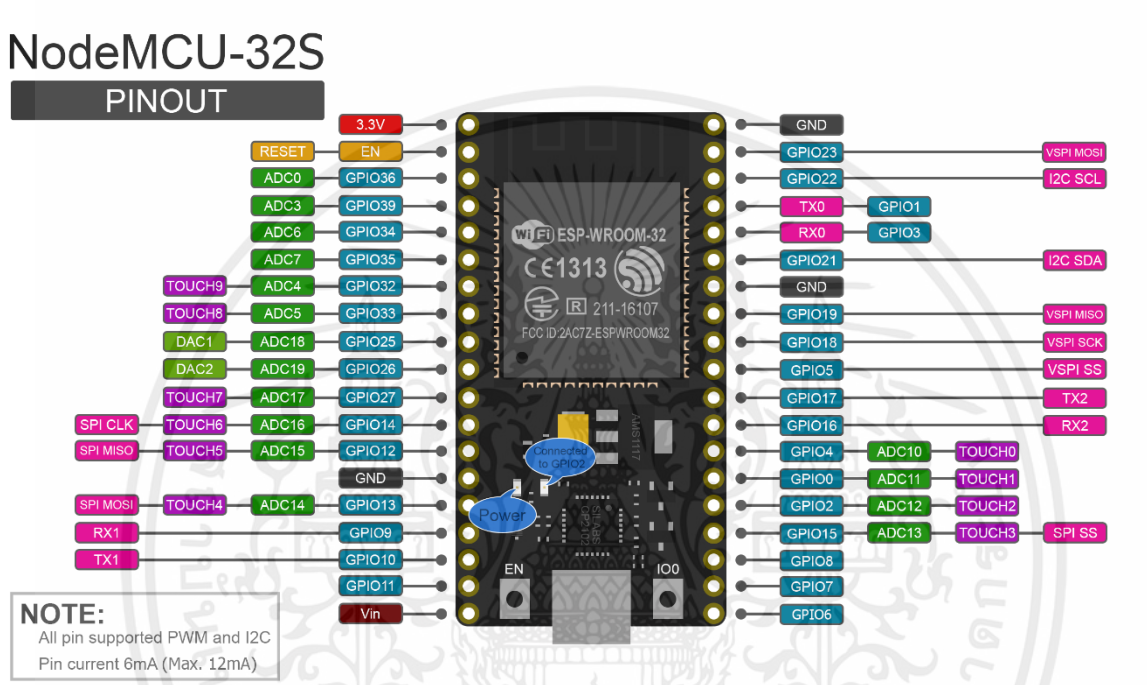
รูปที่ 2.19 บอร์ด NodeMCU-32S ด้านหลัง

(ที่มา : ESP32 **เบื้องต้น :: บทที่ 2 บอร์ดพัฒนา** ESP32 - ArtronShop **บอร์ดอิเล็กทรอนิกส์** Arduino ESP32 ESP8266 : Inspired by LnwShop.com,2564)

บอร์ด NodeMCU-32S ใช้ชิปแปลง USB เป็น UART เบอร์ CP2102 จากบริษัท Silicon Labs สามารถเข้าโหมดอัพโหลดโปรแกรมแบบอัตโนมัติจัดวงจรแบบ nodemcu มีรอม 4MB (หรือ 32Mbit) ใช้ไอซี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เรกกูเลเตอร์แบบ LDO เบอร์ AMS1117-3.3 รองรับแรงดันเข้าสูงสุด 7V และจ่ายกระแสได้สูงสุด 700mA หลอด LED สีแดงแสดงสถานะการจ่ายไฟเลี้ยงให้โมดูล ESP32S / ESP-WROOM-32 และหลอด LED สีฟ้าเชื่อมต่อกับ GPIO2 สวิตซ์กดติดปล่อยดับจำนวน 2 ตัว เชื่อมต่อกับขา CHP_PU ของโมดูล ESP32S / ESP-WROOM-32 ใช้สำหรับรีเซ็ตบอร์ด จำนวน 1 ตัว และเชื่อมต่อกับ GPIO0 เพื่อใช้เข้าโหมดโปรแกรมด้วยตัวเองอีก 1 ตัว ใช้พลังงานไฟฟ้าและสื่อสารผ่านพอร์ต MicroUSB มีขาต่อใช้งานทั้งหมด 38 ขา เรียงขาแบบเดียวกับบอร์ด ESP32-DevKitC เมื่อนำไปเสียบลงโพรโต้บอร์ด



รูปที่ 2.20 Pinout ของบอร์ด NodeMCU-32S

(ที่มา : ESP32 เบื้องต้น :: บทที่ 2 บอร์ดพัฒนา ESP32 - ArtronShop บอร์ดอิเล็กทรอนิกส์ Arduino ESP32 ESP8266 : Inspired by LnwShop.com,2564)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.3.4 WEMOS LOLIN32

รูปที่ 2.21 บอร์ด WEMOS LOLIN32 ด้านหน้าและด้านหลัง

(ที่มา : ESP32 **เบื้องต้น :: บทที่** 2 **บอร์ดพัฒนา** ESP32 - ArtronShop **บอร์ดอิเล็กทรอนิกส์** Arduino ESP32 ESP8266 : Inspired by LnwShop.com,2564)

เป็นบอร์ดจากผู้ผลิต WeMos ที่มีชื่อเสียงด้านการทำบอร์ดพัฒนา ESP8266 ที่มีขนาดเล็ก และถูก รวมถึงมีบอร์ดเสริมต่าง ๆ ช่วยให้สามารถใช้งานได้ง่ายขึ้น ต่อจากการทำบอร์ดพัฒนา ESP8266 แบรนด์ WeMos ก็ได้หันมาทำบอร์ดพัฒนา ESP32 เป็นครั้งแรก ตัวบอร์ดรองรับการชาร์จ ใช้พลังงานไฟฟ้าจาก แบตเตอรี่ลิโพผ่านคอนเนคเตอร์แบบ JST และแตกต่างจากทุกบอร์ดตรงที่ใช้ชิปไอซีแปลง USB เป็น UART เบอร์ CP2104 จากบริษัท Silicon Labs ที่มีขนาดเล็กกว่าชิปไอซี CP2102 และตามสไตล์ของ WeMos คือไม่ มีปุ่มควบคุม GPIO0 ทำให้ไม่สามารถกดเข้าโหมดอัพโหลดโปรแกรมด้วยตนเองได้ บอร์ด WEMOS LOLIN32 ใช้โมดูล ESP-WROOM-32 มีรอม 4MB (หรือ 32Mbit) ใช้ไอซีเรกกูเลเตอร์แบบ LDO ไม่ทราบเบอร์ หลอด LED แสดงสถานะการชาร์จแบตเตอรี่ และหลอด LED เชื่อมต่อกับ GPIO5 สวิตซ์กดติดปล่อยดับเชื่อมต่อกับ ขา CHP_PU ของโมดูล ESP-WROOM-32 ใช้สำหรับรีเซ็ตบอร์ด ชิปไอซีจัดการ ๆ ชาร์จแบตเตอรี่ถูกตั้งไว้ให้ จ่ายกระแสชาร์จที่ 500mA ใช้พลังงานไฟฟ้าและสื่อสารผ่านพอร์ต MicroUSB มีขาต่อใช้งานทั้งหมด 40 ขา

2.3 ESP32 DEVKIT V1

ESP32 DEVKIT V1 เป็นไมโครคอนโทรลเลอร์หรือโมดูลของ ESP32 ตัวโมดูลประกอบด้วย Wi-Fi กับ Bluetooth 4.2 และ Touch/Temp Sensor ทำงานแบบ Dual Core ที่ความเร็ว 160Mhz มีขา 30 ขา มี SRAM 512K หน่วยความจำ Flash สำหรับอัพโหลดโปรแกรมขนาด 16M ความละเอียดในการอ่านค่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ADC 12 Bit สามารถเขียนโปรแกรม ผ่าน Arduino IDE เหมือน Arduino ได้ และรองรับไลบารี่ส่วนใหญ่ของ Arduino

รูปที่ 2.22 บอร์ด ESP32 DEVKIT V1

รูปที่ 2.23 บอร์ด ESP32 DEVKIT V1 pinout

(ที่มา : ESP32 DEVKIT V1 Board (Wi-Fi and Bluetooth) (econo-robot.com),2565)

2.4 Arduino IDE

โปรแกรม Arduino IDE เป็นโปรแกรมที่ใช้สำหรับการเขียนโปรแกรมควบคุมและสามารถทำการแปลงไฟล์ดังกล่าวเพื่อนำไปอัพโหลดลงยังบอร์ด Arduino และไมโครคอนโทรลเลอร์ตระกูล ESP โดยคำว่า Arduino IDE นั้นมาจากคำว่า Arduino คือ ชื่อไมโครคอนโทรลเลอร์ Arduino และ IDE ย่อมาจาก Integrated Development Environment คือ ส่วนเสริมของระบบการพัฒนาหรือตัวช่วยต่างๆที่จะคอยช่วยเหลือคนที่พัฒนา Application เพื่อเสริมให้เกิดความรวดเร็ว ถูกต้อง แม่นยำ ตรวจสอบระบบที่จัดทำได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทำให้การพัฒนางานต่างๆเร็วมากขึ้น สำหรับ Arduino IDE จะมีวิธีการใช้โปรแกรมอยู่ 2 แบบ คือ แบบ online IDE และแบบ desktop IDE

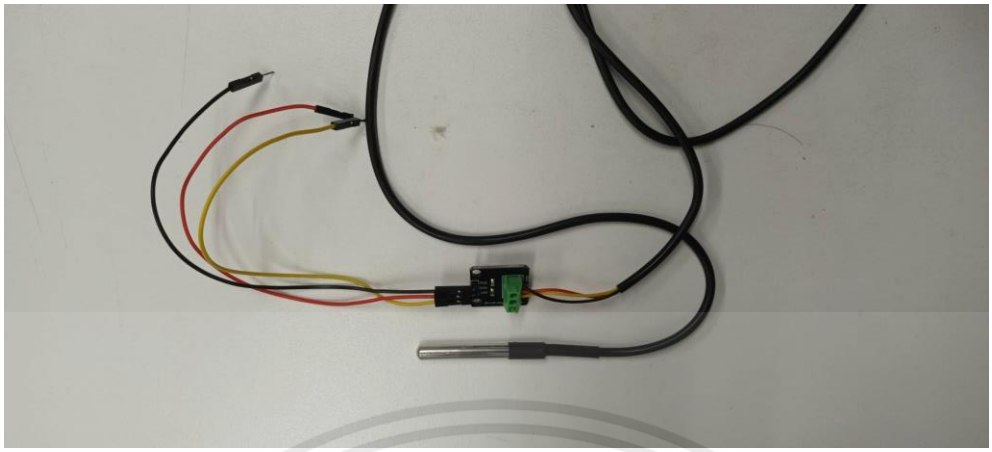
1.โปรแกรมแบบ online IDE (Arduino Web Editor) จะเป็นการเขียนโปรแกรมผ่านเว็บไซค์ โดยข้อมูลต่างๆ ที่เราทำการเขียนขึ้นมา จะถูกนำไปเก็บไว้บน Cloud ซึ่งทำให้สะดวกในการใช้งานที่ไหนก็ได้ นอกจากนั้นเรายังไม่จำเป็นต้องอัพเดทโปรแกรมหรือไลบรารี่

2.โปรแกรมแบบ desktop IDE เป็นโปรแกรมแบบ offline ต้องทำการดาวน์โหลดโปรแกรม ซึ่งจะมีโปรแกรมให้เลือกตามการใช้งาน OS ของเรา

รูปที่ 2.24 บอร์ด หน้าตาของโปรแกรม Arduino IDE แบบ desktop

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5 DS18B20 Temperature Sensor



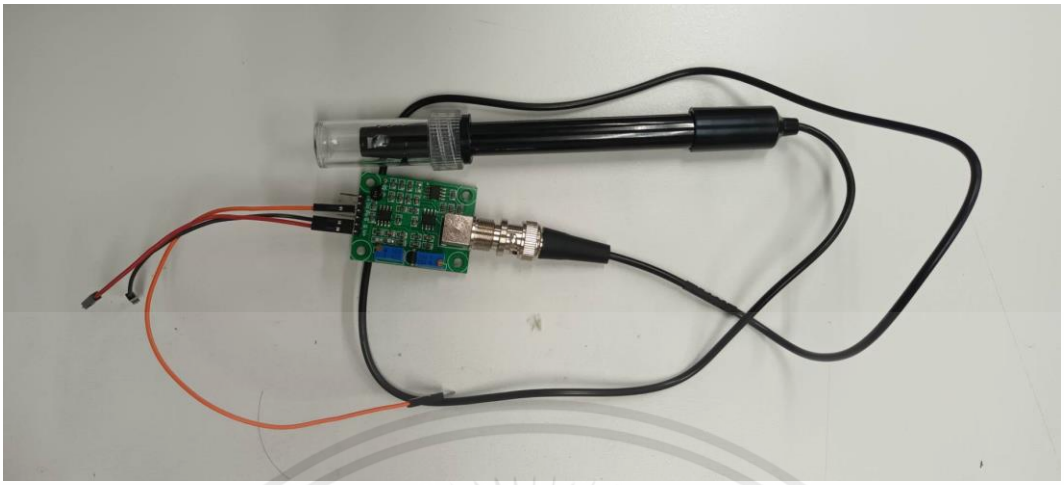
รูปที่ 2.25 DS18B20 Temperature Sensor

DS18B20 Temperature Sensor คือ เซ็นเซอร์สำหรับวัดอุณหภูมิ สามารถวัดอุณหภูมิได้ตั้งแต่ -55 ℃ ถึง 125 ℃ มีค่าความคลาดเคลื่อน 0.5 ℃ ใช้ไฟเลี้ยง 3-5.5 V กินกระแสไฟฟ้า 1 mA มีความละเอียด 9-12 bit ข้อดีของ DS18B20 Temperature Sensor คือมีรูปแบบแพ็คเกจของตัวเซ็นเซอร์หลากหลายรูปแบบ เช่น รูปแบบไอซี รูปแบบหัวเซ็นเซอร์แบบกันน้ำทำให้สามารถใช้วัดอุณหภูมิในของเหลวได้ การใช้งาน DS18B20 Temperature Sensor จะใช้ขา D เพียงขาเดียวในการรับ/ส่งข้อมูล เนื่องจากเซ็นเซอร์ตัวนี้สามารถใช้การสื่อสารแบบ One-Wire communication ได้

การสื่อสารแบบ One-Wire เป็นระบบบัสข้อมูลแบบ Half-duplex สามารถสื่อสารได้ 2 ทิศทาง แต่ไม่สามารถรับและส่งข้อมูลพร้อมกันในช่วงเวลาเดียวกันได้ ระบบบัสมีการทำงานเป็นแบบ Master/Slave โดยอุปกรณ์ Master จะเป็นตัวควบคุมสถานะ และจังหวะการรับส่งของบัสข้อมูล ในขณะที่อุปกรณ์ Slave จะทำงานตามการควบคุมของอุปกรณ์ Master เท่านั้น สำหรับ DS18B20 นั้นจะทำตัวเป็นสเลฟ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

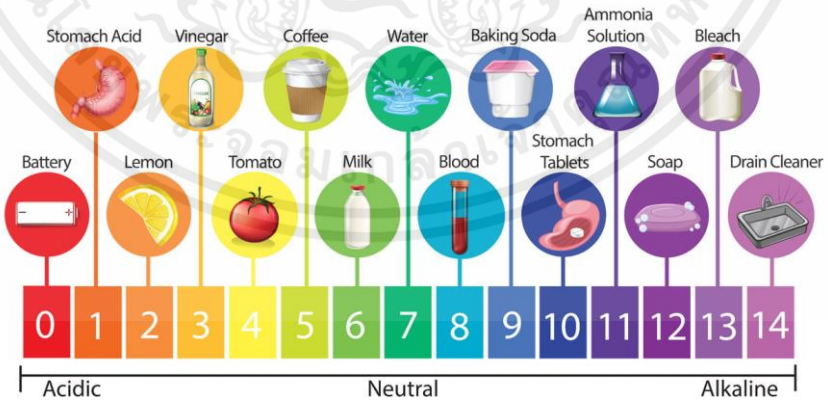
2.6 pH4502 รุ่น E201-C



รูปที่ 2.26 pH4502 รุ่น E201-C

pH4502 รุ่น E201-C เป็นเซ็นเซอร์วัดค่า pH ในน้ำโดยใช้ probe ซึ่ง probe นี้วัดค่า pH และ ORP (Oxidation-Reduction potential) โดย pH probe สามารถวัดค่าความเป็นกรดกับเบสได้ 0-14 และ ORP probe จะคืนค่าแรงดันไฟฟ้าตามสัดส่วนของแนวโน้มการได้รับหรือสูญเสียอิเล็คตรอนของสารละลายจากสารอื่นๆ สำหรับ pH4502 รุ่น E201-C จะใช้ไฟเลี้ยง 5 V กินกระแสไฟฟ้า 5-10 mA กินพลังงานน้อยกว่า 0.5 W มี Response time น้อยกว่า 5 วินาที มี Stability time น้อยกว่า 60 วินาที

สำหรับการวัดค่า pH ในน้ำ มีเกณฑ์ค่าความเป็นกรด-เบส ดังนี้

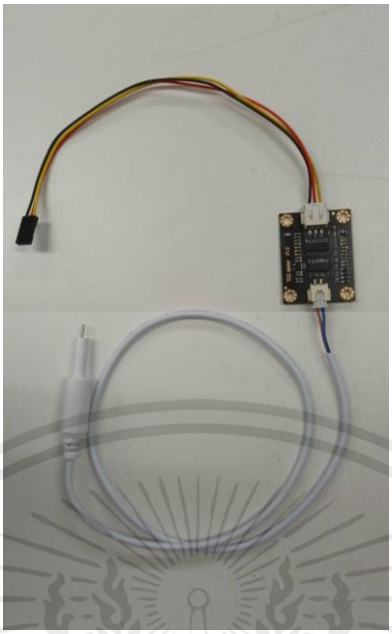


รูปที่ 2.27 ตารางค่าความเป็นกรดเบส

(ที่มา : รู้จักค่า ph คือความเป็นกรด-ด่างของสารละลาย ของเหลวและน้ำ (neonics.co.th),2564)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7 KS0429 keyestudio TDS Meter V1.0



รูปที่ 2.28 KS0429 keyestudio TDS Meter V1.0

Keyestudio TDS เป็นเซ็นเซอร์ที่ใช้วัดค่า TDS (Total dissolved solids) หรือก็คือค่าปริมาณของแข็งที่ละลายได้ทั้งหมดหรือการวัดปริมาณรวมที่ละลายของสารอนินทรีย์และอินทรีย์ทั้งหมดที่อยู่ในของเหลวในรูปแบบแขวนลอยแบบโมเลกุล แตกตัวเป็นไอออน ค่า TDS แสดงในหน่วยส่วนในล้านส่วน (ppm) โดย Keyestudio TDS ใช้ไฟเลี้ยง 3.3-5.5 V กินกระแสไฟฟ้า 3-6 mA สามารถวัดค่า TDS ได้ 0-1000 ppm มีความแม่นยำ 10% ที่อุณหภูมิ 25 °C

โดยทั่วไปความเข้มข้นของของแข็งที่ละลายทั้งหมดคือผลรวมของไอออนบวกและไอออนลบในน้ำส่วนต่อล้านส่วน (ppm)

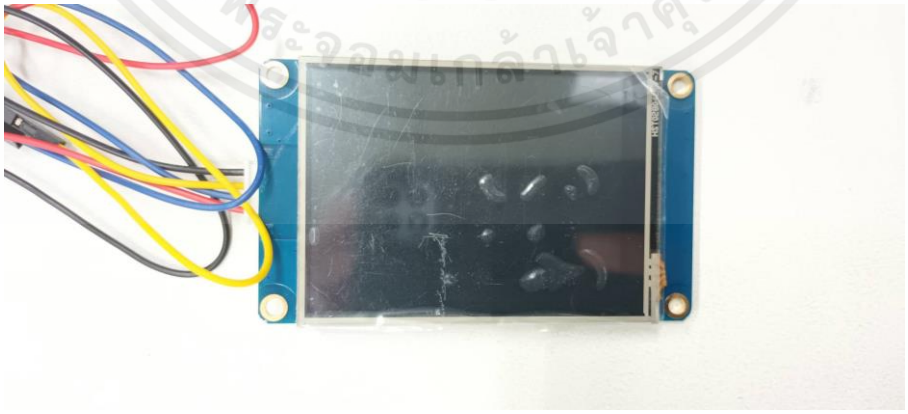
มีการใช้แนวทางต่างๆ เพื่อกำหนดระดับ TDS ที่ควรเป็น สำหรับน้ำดื่มคุณภาพสูงระดับ TDS ควรอยู่ที่ 300 มก./ลิตร หรือน้อยกว่า หากค่าที่อ่านได้อยู่ระหว่าง 300-600 มก./ลิตร คุณภาพน้ำจะดี ค่าใดๆ ระหว่าง 600-900 ถือว่าไม่ดี ในขณะที่การอ่านค่าใดๆ ที่มากกว่า 1,200 มก./ลิตร ถือว่าไม่เป็นที่ยอมรับสำหรับน้ำดื่ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.29 ตารางค่า TDS ที่ยอมรับ

(ที่มา : 3in1 เครื่องทดสอบคุณภาพน้ำ TDS /EC /อุณหภูมิ ตรวจความสะอาดของน้ำ (bbpowershop.com),2564)

2.8 Nextion



รูปที่ 2.30 จอ Nextion

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Nextion เป็น Human Machine Interface (HMI) ชนิดหนึ่ง (HMI คือ การนำระบบคอมพิวเตอร์มาเป็นอุปกรณ์ที่ใช้ในการติดต่อระหว่างผู้ใช้งานกับเครื่องจักร เพื่อควบคุมและเป็นจอแสดงผล) Nextion เป็นแพลตฟอร์มสำหรับการอินเตอร์เฟสร่วมกันระหว่างหน้าจอแสดงผลที่เป็นระบบสัมผัส (Touch screen) กับไมโครคอนโทรลเลอร์ผ่านระบบสื่อสารอนุกรม สำหรับ Nextion จะใช้ซอฟต์แวร์ Nextion Editor ในการออกแบบ

2.8.1 Nextion Editor

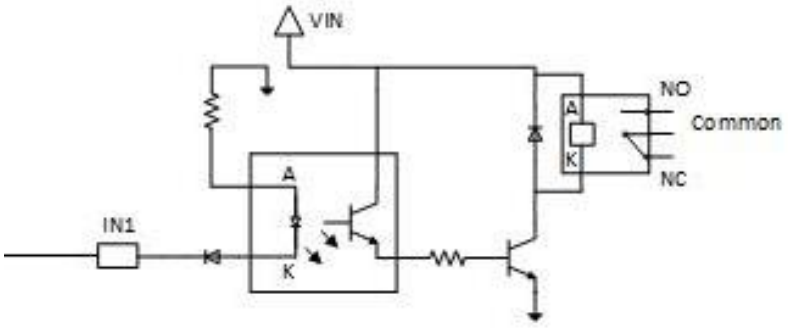
Nextion Editor เป็นโปรแกรมเชิงวัตถุ (Object Oriented Paradigm) สำหรับใช้ในการออกแบบ GUI (Graphical User Interface) ให้แก่บอร์ด Nextion ซึ่งได้จัดเตรียมออบเจ็กต์ (Object) สำหรับการออกแบบหน้าจอไว้อย่างครบถ้วน เช่น ปุ่มกด (Button), ข้อความ (Text), ไทเมอร์(Timer) เป็นต้น

2.9 Relay module

รูปที่ 2.31 Relay module

หลักการทำงานวงจรของ Relay module คือ การเปิด-ปิดวงจรด้วยแม่เหล็กไฟฟ้า เมื่อขดลวดเหนี่ยวนำมีกระแสไหลผ่านจะมีคุณสมบัติเป็นแม่เหล็ก สามารถส่งแรงผลักหรือดูดเพื่อเปลี่ยนตำแหน่งสวิตช์ได้ ดังนั้นใน Relay Module จะประกอบด้วยสองวงจรดังตัวอย่างในรูปข้างล่าง วงจรทางด้านซ้ายเป็นวงจรเพื่อเหนี่ยวนำขดลวดซึ่งจะต่ออยู่กับบอร์ดควบคุม วงจรด้านขวาเป็นวงจรของอุปกรณ์ที่เราต้องการขับ โดยมีสวิตช์เปิดปิดวงจรตามแรงดูดของขดลวด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.32 วงจรภายใน Relay module

หน้าสัมผัสของสวิตช์มี 2 ชนิด คือ

1.หน้าสัมผัสปกติเปิดหรือ NO (Normally Open) หมายถึง หน้าสัมผัสที่เปิดในภาวะขดลวดไม่เหนี่ยวนำ

2.หน้าสัมผัสปกติปิดหรือ NC (Normally Closed) หมายถึง หน้าสัมผัสที่ปิดในภาวะขดลวดไม่เหนี่ยวนำ

โดยทั่วไปแล้วหน้าสัมผัส NO คือ ฝั่งที่ทำให้กระแสครบวงจรในฝั่งอุปกรณ์ ดังแสดงในรูปตัวอย่าง นั่นหมายถึงจะต้องมีการจ่ายไฟให้ขดลวดในวงจรด้านซ้าย เพื่อดูดสวิตช์ในวงจรฝั่งขวามาที่หน้าสัมผัส NO วงจรจึงจะปิด และวงจรด้านขวาจะทำงาน การเปิดปิดกระแสผ่านขดลวดในวงจรฝั่งซ้าย กระทำผ่านการควบคุมไฟเลี้ยงทรานซิสเตอร์ เมื่อทำให้เกิดการจ่ายไฟเลี้ยง (VCC) ทรานซิสเตอร์จะนำกระแส ทำให้วงจรด้านขดลวดปิด และขดลวดจะทำหน้าที่เป็นแม่เหล็ก ซึ่งบอร์ด Arduino จะสามารถเข้ามาควบคุม Relay ได้ที่จุด In1 กล่าวคือถ้าส่งลอจิก High จะไม่มีการจ่ายไฟเลี้ยง เนื่องจากไม่มีความต่างศักย์ ในทางกลับกัน หากส่งลอจิก Low จะทำให้วงจรปิด และสวิตช์จะเปลี่ยนทิศทาง ดังนั้นวงจร Relay ลักษณะนี้จึงเป็นแบบ Active Low

2.10 Motor

มอเตอร์ไฟฟ้า (อังกฤษ: electric motor) เป็นอุปกรณ์ไฟฟ้าที่แปลงพลังงานไฟฟ้าเป็นพลังงานกล การทำงานปกติของมอเตอร์ไฟฟ้าส่วนใหญ่เกิดจากการทำงานร่วมกันระหว่างสนามแม่เหล็กของแม่เหล็กในตัวมอเตอร์ และสนามแม่เหล็กที่เกิดจากกระแสในขดลวดทำให้เกิดแรงดูดและแรงผลักของสนามแม่เหล็กทั้งสอง ในการใช้งานตัวอย่างเช่น ในอุตสาหกรรมการขนส่งใช้มอเตอร์ฉุดลาก เป็นต้น นอกจากนั้นแล้วมอเตอร์ไฟฟ้ายังสามารถทำงานได้ถึงสองแบบ ได้แก่ การสร้างพลังงานกล และ การผลิตพลังงานไฟฟ้า

มอเตอร์ไฟฟ้าถูกนำไปใช้งานที่หลากหลายเช่น พัดลมอุตสาหกรรม เครื่องเป่า ปั๊ม เครื่องมือเครื่องใช้ในครัวเรือน และดิสก์ไดรฟ์ มอเตอร์ไฟฟ้าสามารถขับเคลื่อนโดยแหล่งจ่ายไฟกระแสตรง (DC) เช่น จากแบตเตอรี่, ยานยนต์หรือวงจรเรียงกระแส หรือจากแหล่งจ่ายไฟกระแสสลับ (AC) เช่น จากไฟบ้าน อินเวอร์เตอร์ หรือ เครื่องปั่นไฟ มอเตอร์ขนาดเล็กอาจจะพบในนาฬิกาไฟฟ้า มอเตอร์ทั่วไปที่มีขนาดและ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คุณลักษณะมาตรฐานสูงจะให้พลังงานกลที่สะดวกสำหรับใช้ในอุตสาหกรรม มอเตอร์ไฟฟ้าที่ใหญ่ที่สุดใช้สำหรับการใช้งานลากจูงเรือ และ การบีบอัดท่อส่งน้ำมันและปั๊มป์สูบจัดเก็บน้ำมันซึ่งมีกำลังถึง 100 เมกะวัตต์ มอเตอร์ไฟฟ้าอาจจำแนกตามประเภทของแหล่งที่มาของพลังงานไฟฟ้าหรือตามโครงสร้างภายในหรือตามการใช้งานหรือตามการเคลื่อนไหวของเอาต์พุต และอื่น ๆ

อุปกรณ์เช่นขดลวดแม่เหล็กไฟฟ้าและลำโพงที่แปลงกระแสไฟฟ้าให้เป็นการเคลื่อนไหว แต่ไม่ได้สร้างพลังงานกลที่ใช้งานได้ จะเรียกถูกว่า actuator และ transducer ตามลำดับ คำว่ามอเตอร์ไฟฟ้านั้น ต้องใช้สร้างแรงเชิงเส้น(linear force) หรือ แรงบิด(torque) หรือเรียกอีกอย่างว่า หมุน (rotary) เท่านั้น

2.11 Firebase

Firebase คืออะไร – เป็นหนึ่งในผลิตภัณฑ์ของ Google โดย Firebase คือ Platform ที่รวบรวมเครื่องมือต่าง ๆ สำหรับการจัดการในส่วนของ Backend หรือ Server side ซึ่งทำให้สามารถ Build Mobile Application ได้อย่างมีประสิทธิภาพ และยังลดเวลาและค่าใช้จ่ายของการทำ Server side หรือการวิเคราะห์ข้อมูลให้อีกด้วย โดยมีทั้งเครื่องมือที่ฟรี และเครื่องมีที่มีค่าใช้จ่าย

Firebase ก่อตั้งขึ้นในปี 2011 โดยแอนดรูลีและเจมส์ เทมปลิน สินค้าเริ่มต้น Firebase ป็นฐานข้อมูลเรียลไทม์ซึ่งมี API ที่ช่วยให้นักพัฒนาในการจัดเก็บและซิงค์ข้อมูล โดย Google Firebase 2.0 กูเกิ้ลได้ซื้อกิจการ Firebase และมีการพัฒนาให้สามารถ จากบริการ backend เก็บข้อมูลอย่างเดียว มาเป็นแพลตฟอร์ม ครบวงจรสำหรับนักพัฒนาแอป รองรับบริการแทบทุกอย่างที่นักพัฒนาแอพป้องใช้งาน

Firebase มีอะไรให้ใช้บ้าง

Firebase มีบริการให้ใช้หลายอย่าง สามารถแบ่งเป็นหมวดหมู่ดังนี้

Build Better Apps

1.Cloud Firestore – จัดเก็บและซิงค์ข้อมูลระหว่างผู้ใช้และอุปกรณ์ในระดับโลกโดยใช้ฐานข้อมูล NoSQL ที่โฮสต์บนคลาวด์ Cloud Firestore ให้การซิงโครไนซ์แบบสดและการสนับสนุนออฟไลน์พร้อมกับการสืบค้นข้อมูลที่มีประสิทธิภาพ การผสานรวมกับผลิตภัณฑ์ Firebase อื่น ๆ ช่วยให้คุณสร้างแอปแบบไร้เซิร์ฟเวอร์ได้อย่างแท้จริง

2.Authentication – จัดการผู้ใช้ของคุณด้วยวิธีที่ง่ายและปลอดภัย Firebase Auth มีหลายวิธีในการตรวจสอบสิทธิ์รวมถึงอีเมลและรหัสผ่านผู้ให้บริการบุคคลที่สามเช่น Google หรือ Facebook และใช้ระบบบัญชีที่คุณมีอยู่โดยตรง สร้างอินเทอร์เฟซของคุณเองหรือใช้ประโยชน์จากโอเพ่นซอร์ส UI ที่ปรับแต่งได้อย่างเต็มที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.Hosting – ลดความซับซ้อนของเว็บโฮสติ้งของคุณด้วยเครื่องมือที่สร้างขึ้นเฉพาะสำหรับเว็บแอปสมัยใหม่ เมื่อคุณอัปโหลดเนื้อหาเว็บของคุณเราจะส่งเนื้อหาเหล่านั้นไปยัง CDN ทั่วโลกของเราโดยอัตโนมัติและมอบใบรับรอง SSL ฟรีเพื่อให้ผู้ใช้ของคุณได้รับประสบการณ์ที่ปลอดภัยเชื่อถือได้และมีเวลาแฝงต่ำไม่ว่าจะอยู่ที่ใดก็ตาม

4.Realtime Database – Realtime Database คือฐานข้อมูลดั้งเดิมของ Firebase เป็นโซลูชันที่มีประสิทธิภาพและมีเวลาแฝงต่ำสำหรับแอปบนอุปกรณ์เคลื่อนที่ที่ต้องการสถานะการซิงค์ระหว่างไคลเอนต์แบบเรียลไทม์ เราขอแนะนำ Cloud Firestore แทน Realtime Database สำหรับนักพัฒนาส่วนใหญ่ที่เริ่มโปรเจ็กต์ใหม่

Improve app quality

1.Crashlytics – ลดเวลาในการแก้ไขปัญหาของคุณด้วยการเปลี่ยนข้อขัดข้องจากหิมะถล่มให้เป็นรายการปัญหาที่จัดการได้ รับข้อมูลเชิงลึกที่ชัดเจนและนำไปปฏิบัติได้ว่าปัญหาใดที่ต้องจัดการก่อนโดยเห็นผลกระทบของผู้ใช้ในแดชบอร์ด Crashlytics การแจ้งเตือนแบบเรียลไทม์จะช่วยให้คุณมีความเสถียรแม้ในขณะเดินทาง Crashlytics เป็นตัวรายงานข้อขัดข้องหลักของ Firebase

2.Performance Monitoring – วินิจฉัยปัญหาประสิทธิภาพของแอปที่เกิดขึ้นบนอุปกรณ์ของผู้ใช้ ใช้การติดตามเพื่อตรวจสอบประสิทธิภาพของบางส่วนของแอปและดูมุมมองสรุปในคอนโซล Firebase อยู่เหนือเวลาเริ่มต้นของแอปและตรวจสอบคำขอ HTTP โดยไม่ต้องเขียนโค้ดใด ๆ

3.Test Lab – เรียกใช้การทดสอบอัตโนมัติและกำหนดเองสำหรับแอปของคุณบนอุปกรณ์เสมือนและจริงที่โฮสต์โดย Google ใช้ Firebase Test Lab ตลอดวงจรการพัฒนาของคุณเพื่อค้นหาจุดบกพร่องและความไม่สอดคล้องกันเพื่อให้คุณสามารถนำเสนอประสบการณ์ที่ยอดเยี่ยมบนอุปกรณ์หลากหลายประเภท

Grow your business

1.Google Analytics – วิเคราะห์คุณลักษณะและพฤติกรรมของผู้ใช้ในแดชบอร์ดเดียวเพื่อทำการตัดสินใจอย่างชาญฉลาดเกี่ยวกับแผนงานผลิตภัณฑ์ของคุณ รับข้อมูลเชิงลึกแบบเรียลไทม์จากรายงานหรือส่งออกข้อมูลเหตุการณ์ดิบไปยัง Google BigQuery สำหรับการวิเคราะห์ที่กำหนดเอง

2.Remote Config – กำหนดวิธีการแสดงผลแอปของคุณสำหรับผู้ใช้แต่ละคน เปลี่ยนรูปลักษณ์เปิดตัวฟีเจอร์ทีละน้อยเรียกใช้การทดสอบ A / B ส่งมอบเนื้อหาที่กำหนดเองให้กับผู้ใช้บางรายหรือทำการอัปเดตอื่น ๆ โดยไม่ต้องปรับใช้เวอร์ชันใหม่ทั้งหมดนี้ทำได้จากคอนโซล Firebase ตรวจสอบผลกระทบของการเปลี่ยนแปลงของคุณและทำการปรับเปลี่ยนในเวลาไม่กี่นาที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.Cloud Messaging – ส่งข้อความและการแจ้งเตือนไปยังผู้ใช้ข้ามแพลตฟอร์มทั้ง Android, iOS และเว็บได้ฟรี สามารถส่งข้อความไปยังอุปกรณ์เดียวกลุ่มอุปกรณ์หรือหัวข้อเฉพาะหรือกลุ่มผู้ใช้ Firebase Cloud Messaging (FCM) ปรับขนาดเป็นแอปที่ใหญ่ที่สุดโดยส่งข้อความหลายแสนล้านข้อความต่อวัน

2.12 Flutter

Flutter เป็นเฟรมเวิร์กโอเพ่นซอร์สที่พัฒนาและสนับสนุนโดย Google นักพัฒนาฟรอนท์เอนด์และฟูลสแตกใช้ Flutter เพื่อสร้างอินเทอร์เฟซผู้ใช้ (UI) ของแอปพลิเคชันสำหรับหลายแพลตฟอร์มด้วยฐานรหัสเดียว Flutter เปิดตัวในปี 2018 รองรับการพัฒนาแอพมือถือเป็นหลัก ตอนนี้ Flutter พร้อมรองรับการพัฒนาแอปพลิเคชันบน 6 แพลตฟอร์มดังนี้: iOS Android เว็บ Windows MacOS และ Linux

Flutter ช่วยลดความยุ่งยากในกระบวนการสร้าง UI ที่สอดคล้องกันและน่าสนใจสำหรับแอปพลิเคชันในหกแพลตฟอร์มที่รองรับ

เนื่องจาก Flutter เป็นเฟรมเวิร์กการพัฒนาข้ามแพลตฟอร์ม อันดับแรกเราจะเปรียบเทียบกับการพัฒนาแบบเนทีฟ จากนั้น เราสามารถเน้นคุณสมบัติที่เป็นเอกลักษณ์ของ Flutter

1.การพัฒนาแอพเนทีฟเทียบกับการพัฒนาแอพข้ามแพลตฟอร์ม

การเข้ารหัสแอปพลิเคชันสำหรับแพลตฟอร์มเฉพาะ เช่น iOS เรียกว่าการพัฒนาแอปแบบเนทีฟ ในทางตรงกันข้าม การพัฒนาแอปข้ามแพลตฟอร์มกำลังสร้างแอปพลิเคชันสำหรับหลายแพลตฟอร์มด้วยโค้ดเบสเดียว

2.การพัฒนาแอปบนมือถือ

เนื่องจากนักพัฒนาโค้ดสำหรับแพลตฟอร์มเฉพาะในการพัฒนาแอพที่มาพร้อมเครื่อง พวกเขาจึงสามารถเข้าถึงฟังก์ชันการทำงานของอุปกรณ์ดั้งเดิมได้อย่างเต็มที่ ซึ่งโดยทั่วไปจะนำไปสู่ประสิทธิภาพและความเร็วที่สูงขึ้นเมื่อเทียบกับการพัฒนาแอปข้ามแพลตฟอร์ม

ในทางกลับกัน หากคุณต้องการเปิดแอปพลิเคชันบนหลายแพลตฟอร์ม การพัฒนาแอปแบบเนทีฟจำเป็นต้องมีโค้ดและนักพัฒนามากขึ้น นอกเหนือจากค่าใช้จ่ายเหล่านี้แล้ว การพัฒนาแอปแบบเนทีฟยังทำให้การเปิดตัวบนแพลตฟอร์มต่างๆ พร้อมกันได้ยากขึ้นด้วยประสบการณ์ผู้ใช้ที่สอดคล้องกัน นี่คือจุดที่เฟรมเวิร์กการพัฒนาแอปข้ามแพลตฟอร์มอย่าง Flutter มีประโยชน์

3.การพัฒนาแอพข้ามแพลตฟอร์ม

การพัฒนาแอพข้ามแพลตฟอร์มช่วยให้นักพัฒนาสามารถใช้ภาษาการเขียนโปรแกรมหนึ่งภาษาและหนึ่ง codebase เพื่อสร้างแอปพลิเคชันสำหรับหลายแพลตฟอร์ม หากคุณกำลังเผยแพร่แอปพลิเคชันสำหรับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลายแพลตฟอร์ม การพัฒนาแอปข้ามแพลตฟอร์มจะมีค่าใช้จ่ายและใช้เวลาน้อยกว่าการพัฒนาแอปแบบเนทีฟ

กระบวนการนี้ยังช่วยให้นักพัฒนาสร้างประสบการณ์ที่สอดคล้องกันมากขึ้นสำหรับผู้ใช้ข้ามแพลตฟอร์ม

แนวทางนี้อาจมีข้อเสียเมื่อเทียบกับการพัฒนาแอปที่มาพร้อมเครื่อง ซึ่งรวมถึงการเข้าถึงฟังก์ชันการทำงานของอุปกรณ์ดั้งเดิมอย่างจำกัด อย่างไรก็ตาม Flutter มีคุณสมบัติที่ทำให้การพัฒนาแอปข้ามแพลตฟอร์มราบรื่นขึ้นและมีประสิทธิภาพสูง

ข้อดีของ Flutter

ต่อไปนี้คือบางวิธีที่ Flutter โดดเด่นในฐานะเฟรมเวิร์กการพัฒนาข้ามแพลตฟอร์ม

1.ประสิทธิภาพใกล้เคียงกับเจ้าของภาษา Flutter ใช้ภาษาการเขียนโปรแกรม Dart และคอมไพล์เป็นรหัสเครื่อง อุปกรณ์โฮสต์เข้าใจรหัสนี้ ซึ่งทำให้มั่นใจได้ถึงประสิทธิภาพที่รวดเร็วและมีประสิทธิภาพ

2.การเรนเดอร์ที่รวดเร็ว สม่ำเสมอ และปรับแต่งได้ แทนที่จะใช้เครื่องมือแสดงผลเฉพาะแพลตฟอร์ม Flutter ใช้ไลบรารีกราฟิกโอเพนซอร์ซของ Google เพื่อแสดงผล UI สิ่งนี้ทำให้ผู้ใช้มีภาพที่สอดคล้องกันไม่ว่าพวกเขาจะใช้แพลตฟอร์มใดในการเข้าถึงแอปพลิเคชันก็ตาม

3.เครื่องมือที่เป็นมิตรกับนักพัฒนา Google สร้าง Flutter โดยเน้นที่ความง่ายในการใช้งาน ด้วยเครื่องมืออย่าง hot reload นักพัฒนาสามารถดูตัวอย่างการเปลี่ยนแปลงโค้ดได้โดยไม่สูญเสียสถานะ เครื่องมืออื่นๆ เช่น เครื่องมือตรวจสอบวิดเจ็ตช่วยให้เห็นภาพและแก้ไขปัญหาด้วยเลย์เอาต์ UI ได้ง่าย

Flutter ใช้ภาษาโปรแกรมอะไร

Flutter ใช้ภาษาโปรแกรมโอเพนซอร์ซ Dart ซึ่งพัฒนาโดย Google ด้วย Dart ได้รับการปรับให้เหมาะสมสำหรับการสร้าง UI และจุดแข็งหลายอย่างของ Dart ถูกใช้ใน Flutter

ตัวอย่างเช่น คุณลักษณะหนึ่งของ Dart ที่ใช้ใน Flutter คือความปลอดภัยที่ไม่มีค่าเสียง ความปลอดภัยของ Dart ที่เป็นโมฆะทำให้ง่ายต่อการตรวจจับจุดบกพร่องทั่วไปที่เรียกว่าข้อผิดพลาดที่เป็นโมฆะ คุณลักษณะนี้ช่วยลดเวลาที่นักพัฒนาใช้ในการบำรุงรักษาโค้ด และทำให้พวกเขามีเวลามากขึ้นในการมุ่งเน้นที่การสร้างแอปพลิเคชันของตน

วิดเจ็ตใน Flutter คืออะไร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ใน Flutter นักพัฒนาสร้างเค้าโครง UI โดยใช้วิดเจ็ต ซึ่งหมายความว่าทุกอย่างที่ผู้ใช้เห็นบนหน้าจอ ตั้งแต่หน้าต่าง แผงปุ่มและข้อความ ล้วนทำจากวิดเจ็ต

วิดเจ็ต Flutter ได้รับการออกแบบมาเพื่อให้นักพัฒนาสามารถปรับแต่งได้อย่างง่ายดาย Flutter บรรลุเป้าหมายนี้ด้วยการจัดองค์ประกอบภาพ ซึ่งหมายความว่าวิดเจ็ตส่วนใหญ่ประกอบด้วยวิดเจ็ตขนาดเล็ก และวิดเจ็ตพื้นฐานส่วนใหญ่มีวัตถุประสงค์เฉพาะ ช่วยให้นักพัฒนาสามารถรวมหรือแก้ไขวิดเจ็ตเพื่อสร้างใหม่ได้

Flutter แสดงวิดเจ็ตโดยใช้เอ็นจิ้นกราฟิกของตัวเองแทนที่จะอาศัยวิดเจ็ตในตัวของแพลตฟอร์ม ด้วยวิธีนี้ ผู้ใช้จะได้สัมผัสกับรูปลักษณ์ที่คล้ายคลึงกันในแอปพลิเคชัน Flutter ข้ามแพลตฟอร์ม วิธีการนี้ยังให้ความยืดหยุ่นแก่นักพัฒนา เนื่องจากวิดเจ็ต Flutter บางตัวสามารถทำหน้าที่ที่วิดเจ็ตเฉพาะแพลตฟอร์มไม่สามารถทำได้

Flutter ยังทำให้ง่ายต่อการใช้วิดเจ็ตที่พัฒนาโดยชุมชน สถาปัตยกรรมของ Flutter รองรับการมีไลบรารีวิดเจ็ตหลายไลบรารี และ Flutter สนับสนุนให้ชุมชนสร้างและดูแลรักษาไลบรารีใหม่

ประเภทของวิดเจ็ต Flutter

Flutter มาพร้อมกับแค็ตตาล็อกวิดเจ็ตมากมายตั้งแต่คุณดาวน์โหลด แคตตาล็อกมี 14 หมวดหมู่ ซึ่งรวมถึงการจัดรูปแบบ Cupertino (วิดเจ็ตสไตล์ iOS) และส่วนประกอบวัสดุ (วิดเจ็ตที่เป็นไปตามหลักเกณฑ์การออกแบบวัสดุของ Google)

Flutter ยังมาพร้อมกับเลย์เอาต์และธีมที่ช่วยให้นักพัฒนาสร้างได้ทันที

Flutter รองรับโดย Google และชุมชนโอเพ่นซอร์สที่ใช้งานบน Reddit, Discord, Slack, Stack Overflow และ Gitter. Google ได้อัปเดต Flutter อย่างสม่ำเสมอตั้งแต่เปิดตัวในปี 2018 รวมถึงการอัปเดต Flutter 3 ในปี 2022 ซึ่งขยายการสนับสนุนที่เสถียรไปยัง macOS และ Linux

เพื่อให้ Flutter ง่ายต่อการเรียนรู้ Google ได้เขียนเอกสารและบทแนะนำมากมายบนเว็บไซต์ของ Flutter เพื่อมีส่วนร่วมกับผู้ใช้ Flutter Google ก็จัดกิจกรรมระดับโลก ส่งเสริมโครงการชุมชน และสนับสนุนความท้าทายของนักพัฒนา เหตุการณ์ที่จะเกิดขึ้นสามารถพบได้บนเว็บไซต์ของ Flutter

ชุมชนของ Flutter ได้สร้างแพ็คเกจของบุคคลที่สามหลายพันรายการและเครื่องมือที่ยอดเยี่ยมที่ปรับปรุงประสบการณ์ของนักพัฒนา ไลบรารีเหล่านี้มีอยู่ที่ pub.dev.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.13 Dart

Dart นั้นเป็นภาษาโปรแกรมที่เอาไว้สำหรับสร้างแอพพลิเคชันบนแพลตฟอร์มที่หลากหลายโดยได้ทั้ง mobile, desktop, server และก็ web สิ่งที่เป็นที่นิยมที่สุดที่ทำให้คนสนใจมาเรียนภาษา Dart กันก็คือเพื่อที่จะเอาไปใช้ร่วมกับ Flutter ที่เป็นเครื่องมือช่วยสร้าง UI ของ Google ซึ่งใช้ได้ทั้งกับ Android และ iOS หรือจะเป็นใน Desktop กับ Web ก็ยังได้

ภาษา Dart นี้ถูกสร้างโดย Google และปล่อยให้ใช้งานแบบ open source ทำให้ทุกคนสามารถนำไปใช้งานได้ฟรีๆ และการที่ Dart ถูกออกแบบมาให้ใช้งานได้ง่ายและมีประสิทธิภาพแบบภาษาเชิงวัตถุอื่นๆ อย่าง Java C# C++ จึงเป็นตัวเลือกภาษาที่น่าสนใจในการศึกษาเป็นภาษาแรกอีกด้วย

รูปที่ 2.33 Dart

2.14 Android Studio

Android Studio เป็นเครื่องมือพัฒนา IDE (ไอ ดี อี) หรือ Integrated Development Environment (อินทิเกรต ดีเวลลอปเม้นท์ (เอนไวรอนเม้นท์) ที่ถูกสร้างขึ้นมาเพื่อการพัฒนาแอนดรอยด์แอปพลิเคชั่น บนพื้นฐานของแนวคิด InteliJ IDEA(อินเทล ไอ เจ ไอดีอีเอ) คล้าย ๆ กับการทำงานของ Eclipse (อีคิปส์)และ Android ADT Plugin (แอนดรอยด์ เอดีที ปลั๊กอิน) และเป็น IDE Tools (ไอ ดี เอ็ม ทูล) ล่าสุดจาก Google (กูเกิ้ล) ไว้พัฒนาโปรแกรม Android (แอนดรอยด์)

สรุป’ยๆ Android Studio คือเป็นระบบปฏิบัติการของอุปกรณ์เคลื่อนที่ ที่เห็นกันชัดๆ นั้นก็คือ โทรศัพท์, แท็บเล็ต เป็นต้น นั้นและครับ ที่พัฒนามาจาก IDE ที่มีผู้ดูแล คือ Google ครับ

การเขียน Android บน Android Studio จะมีขั้นตอนอยู่ 2 ขั้นตอนก็คือ ติดตั้ง Java SDK (จาวา เอสดีเค)และดาวน์โหลด Android Studio มาติดตั้งก็จะสามารถใช้งานได้ทันที่ ละยังรวมไปถึงตัว Emulator อย่างเช่น Genymotion (จีนีโมชั่น) ที่เราต้องโหลดมาติดตั้งเพื่อช่วยในการทดสอบ Project (โปรเจ็ค)ที่เราเขียนอีกด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.34 Android studio

2.15 Visual Studio

ในการทำธุรกิจในปัจจุบัน ได้มีการนำเทคโนโลยีต่างๆ เข้ามาช่วยในการทำงาน เนื่องจากสามารถทำงานได้รวดเร็ว และผิดพลาดน้อย สามารถทำงานได้ดีกว่าพนักงานบางคน ดังนั้นนักธุรกิจหรือกิจการส่วนใหญ่จึงได้นำเทคโนโลยีต่างๆ เข้ามาช่วยในการทำงาน รวมถึงระบบของคอมพิวเตอร์ต่างๆ เพื่อช่วยให้อำนวยความสะดวกในการทำงานให้แก่พนักงาน และลดข้อผิดพลาดต่างๆ ลงไป ซึ่งระบบการทำงานต่างๆ นั้น ถูกพัฒนาขึ้นมาโดยนักโปรแกรมเมอร์ ซึ่งผู้พัฒนาจะต้องรู้จักกับภาษาของคอมพิวเตอร์ หรือพูดคุยกับคอมพิวเตอร์ได้ ซึ่งคอมพิวเตอร์นั้นมีหลายภาษา เช่น C++ , C# , Java, VB, VB.NET, PHP, PYTHON, GROOVY และภาษาอื่นๆ อีกมากมาย ซึ่งภาษาแต่ละภาษาก็มีความสามารถและการใช้งานที่แตกต่างกันไปขึ้นอยู่กับระบบที่ต้องการพัฒนา ในปัจจุบันได้มีเครื่องมือหรือโปรแกรมที่เข้ามาช่วยให้ผู้พัฒนาไม่จำเป็นต้องเขียนภาษาพูดคุยกับคอมพิวเตอร์เองทั้งหมด แต่โปรแกรมจะช่วยพูดคุยกับคอมพิวเตอร์ให้ในระดับหนึ่งแล้ว ซึ่งช่วยอำนวยความสะดวกและลดเวลาการทำงานให้แก่ผู้พัฒนาได้เป็นอย่างมาก ซึ่งโปรแกรมเหล่านี้ก็มีเป็นจำนวนมากเช่นกัน ขึ้นอยู่กับว่าจะใช้ภาษาใดเขียน และโปรแกรมใดที่เหมะสมกับภาษานั้น ซึ่งจะขอแนะนำโปรแกรมอีกตัวหนึ่งที่ชื่อว่า Visual Studio

Visual Studio (วิชวลสตูดิโอ) คือ โปรแกรมตัวหนึ่งที่เป็นเครื่องมือที่ช่วยพัฒนาซอฟต์แวร์และระบบต่างๆ ซึ่งสามารถติดต่อสื่อสารพูดคุยกับคอมพิวเตอร์ได้ในระดับหนึ่งแล้ว แต่ยังไม่สามารถพัฒนาเป็นระบบเองได้ เหมาะสมสำหรับภาษา VB และ VB.NET เนื่องจากไมโครซอฟต์ได้พัฒนาโปรแกรมและภาษาขึ้นมาควบคู่กันเพื่อให้ใช้งานได้ซึ่งกันและกัน ซึ่งนักโปรแกรมเมอร์จะนำเครื่องมือมาใช้ในการพัฒนาต่อยอดให้เกิดเป็นระบบต่างๆ หรือเป็นเว็บไซต์ และแอพพลิเคชั่นต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.35 Microsoft Visual Studio

2.16 Visual Studio Code

Visual Studio Code หรือ VSCode เป็นโปรแกรม Code Editor ที่ใช้ในการแก้ไขและปรับแต่งโค้ด จากค่ายไมโครซอฟท์ มีการพัฒนาออกมาในรูปแบบของ OpenSource จึงสามารถนำมาใช้งานได้แบบฟรี ๆ ที่ต้องการความเป็นมืออาชีพ

ซึ่ง Visual Studio Code นั้น เหมาะสำหรับนักพัฒนาโปรแกรมที่ต้องการใช้งานข้ามแพลตฟอร์ม รองรับการใช้งานทั้งบน Windows, macOS และ Linux สนับสนุนทั้งภาษา JavaScript, TypeScript และ Node.js สามารถเชื่อมต่อกับ Git ได้ นำมาใช้งานได้ง่ายไม่ซับซ้อน มีเครื่องมือส่วนขยายต่าง ๆ ให้เลือกใช้อย่างมากมาก ไม่ว่าจะเป็น 1.การเปิดใช้งานภาษาอื่น ๆ ทั้ง ภาษา C++, C#, Java, Python, PHP หรือ Go 2.Themes 3.Debugger 4.Commands เป็นต้น

ความแตกต่างระหว่าง VSCode และ Visual Studio คือ

- VSCode ได้ทำการตัดในส่วนของ GUI designer ออกไป เหลือแต่เพียงตัว Editor เท่านั้น จึงทำให้ตัวโปรแกรมนั้นค่อนข้างเบากว่า Visual Studio เป็นอย่างมาก

- VSCode สามาถนำมาใช้งานได้ฟรี รองรับการทำงานข้ามแพลตฟอร์ม

สำหรับผู้ที่สนใจใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.36 Visual Studio Code

2.17 พัดลมตู้ปลา (Fan)

พัดลมตู้ปลา คือ พัดลมที่ใช้สำหรับตู้ปลาโดยเฉพาะ กล่าวคือ เป็นพัดลมขนาดเล็กที่มีอุปกรณ์หนีบไว้ข้างตู้ สามารถทำให้อุณหภูมิในตู้ปลาลดลงประมาณ 3-4 องศา ใช้ไฟเลี้ยง 12 V

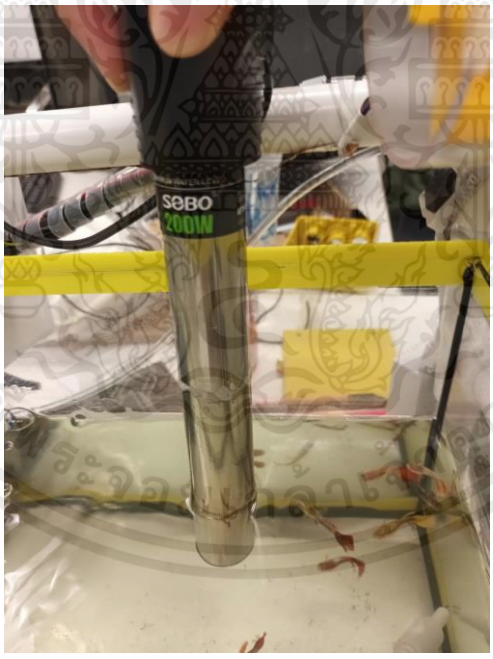
รูปที่ 2.37 พัดลมตู้ปลา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.18 ฮีทเตอร์สำหรับตู้ปลา (Aquarium heater)

ฮีทเตอร์สำหรับตู้ปลาเป็นอุปกรณ์ที่ใช้ในงานอดิเรกในการเลี้ยงปลาเพื่อให้อุณหภูมิของน้ำในตู้ปลาอุ่นขึ้น พิพิธภัณฑ์สัตว์น้ำจืดและสัตว์น้ำทะเลในเขตร้อนส่วนใหญ่ได้รับการดูแลที่อุณหภูมิตั้งแต่ 22 ถึง 30 °C (71-86 °F) ประเภทรวมถึงเครื่องทำความร้อนแบบจุ่มแก้วและเครื่องทำความร้อนใต้ชั้นหิน นอกจากนี้ยังมีเสื่อทำความร้อนที่อาจวางไว้ใต้ตู้ปลา

โดยทั่วไป เครื่องทำความร้อนในตู้ปลาจะเป็นเครื่องทำความร้อนแบบจุ่ม เครื่องทำความร้อนเหล่านี้ประกอบด้วยท่อแก้วที่มีองค์ประกอบความร้อนพันรอบแผ่นเซรามิกหรือแก้ว เครื่องทำความร้อนแบบแช่แก้วบางชนิดมีทรายอยู่ด้วย และส่วนใหญ่ แต่ไม่ใช่ทั้งหมดที่สามารถแช่ได้ทั้งหมด หลอดแก้วยังมีเทอร์โมสตัทแบบปรับได้ซึ่งจะเปิดองค์ประกอบความร้อนเพื่อรักษาอุณหภูมิที่ต้องการ เทอร์โมสตัทนี้มักเป็นแถบสองโลหะ เนื่องจากแถบประกอบด้วยโลหะสองชนิด โลหะจะขยายตัวในอัตราที่ต่างกันเมื่ออุณหภูมิสูงขึ้น ทำให้แถบโค้งงอ แถบนี้นำกระแสไฟฟ้า แต่แบ่งกระแสไฟฟ้าที่อุณหภูมิที่ถูกต้อง แถบ bimetallic นี้สามารถปรับการตั้งค่าได้เทอร์โมสตัทขั้นสูงอาจใช้เทคโนโลยีไมโครชิป มักจะมีไฟดวงเล็กในตัวทำความร้อนเพื่อระบุการทำงาน



รูปที่ 2.38 ฮีทเตอร์สำหรับตู้ปลา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.19 ไฟ LED (Strip LED Light)

ไฟเส้น LED เป็นที่นิยมอย่างมากในปัจจุบัน เนื่องจากมีค่า IP หลากหลายให้เลือกใช้งานตั้ง IP20 ไปจนถึง IP68 โดยเฉพาะอย่างยิ่งในงานตกแต่งภายในและภายนอก การติดตั้งใต้น้ำก็เป็นที่นิยมมากไม่แพ้กัน ด้วยฟังก์ชั่นที่หลากหลายทั้งยังติดตั้งได้ง่าย รวมถึงสามารถหลบซ่อนสายตาจากผู้คนได้อย่างดีจึงเหมาะใช้เป็นไฟซ่อนใต้ฝ้า เพดาน ตามขอบประตู หน้าต่าง ซ่อนใต้ท้องรถหรือใช้ตกแต่งเฟอร์นิเจอร์ก็ได้เช่นกัน

ไฟชนิดนี้มีลักษณะเป็นเส้นยาวคล้ายริบบิ้น มันจึงมีชื่อเรียกว่า LED Ribbon ด้วยคุณสมบัติเฉพาะตัวสามารถบิดงอได้อย่างอิสระ(ตามองศาโค้งของแต่ละรุ่น) ติดตั้งได้ง่ายแม้ในพื้นที่แคบ ดัดโค้งปรับให้เข้ากับโครงสร้างพื้นที่ที่ต้องการติดตั้งได้ง่าย ให้แสงสว่างที่ต่อเนื่องไม่เกิดเงา ด้วยข้อดีทั้งหมดนี้ ปัจจุบันเราจึงเห็นไฟเส้น LED ในแทบทุกพื้นที่ โดยเฉพาะตามห้างสรรพสินค้าและสถานบริการต่าง ๆ

รูปที่ 2.39 รูปภาพไฟ LED Strip

2.20 ปั๊มน้ำ (Water pump)

ปั๊มน้ำเป็นอีกหนึ่งอุปกรณ์สำหรับผู้นิยมเลี้ยงปลาสวยงามใช้เป็นเครื่องไฟฟ้าชิ้นหลักในตู้หรือบ่อปลากันครับ โดยมีลักษณะเป็นกึ่งสี่เหลี่ยม มีช่องให้น้ำเข้า และน้ำออกอยู่คนละด้านกัน เวลาใช้งานตัวเครื่องจะต้องจมอยู่ในน้ำทั้งหมด ซึ่งปั๊มน้ำที่มีมาตรฐานจะมีระบบป้องกันไฟฟ้ารั่วเป็นอย่างดี

โดยเป็นการเพิ่มแรงดันของน้ำในไหลเร็วแล้วแรงมากยิ่งขึ้น ซึ่งเราสามารถนำไปใช้กับเครื่องกรองน้ำ เพิ่มปริมาณอ๊อกซิเจนในน้ำ หรือเพิ่มการหมุนเวียนของน้ำได้เป็นอย่างดี เรายังสามารถใช้ปั๊มน้ำนี้ประยุกต์ใช้กับระบบน้ำพุ น้ำตก แถมยังปั๊มน้ำออกจากบ้านเราตอนน้ำท่วมได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.40 รูปภาพปั๊มน้ำสำหรับตู้ปลา

2.21 เครื่องกรองน้ำ (Filter)

การกรองน้ำในตู้ปลามีความสำคัญมาก เพราะการเลี้ยงปลาสวยงามผู้เลี้ยงต้องการให้มีความสวยงามมากที่สุด น้ำควรจะต้องใส ไม่มีเศษอาหาร ตะกอน หรือสิ่งขับถ่ายของปลามาลอยรบกวนสายตา อีกทั้งยังช่วยกำจัดสิ่งไม่ต้องการดังกล่าวออกจากตู้ปลาได้ด้วย ทำให้น้ำมีคุณภาพดี ส่งผลให้ปลามีสุขภาพดีด้วย การกรองน้ำจึงมีความจำเป็นในระบบการเลี้ยงปลาสวยงาม เครื่องกรองน้ำหรืออุปกรณ์ที่ใช้ในการกรองน้ำมีหลายชนิด

ระบบกรองน้ำแบบหม้อกรองในตู้ หรือ Box Filter เป็นระบบกรองน้ำที่สามารถ ติดตั้งได้สะดวก ง่าย และรวดเร็วที่สุด นิยมใช้กันมากกับการเลี้ยงปลาสวยงามในตู้ขนาดเล็ก

อุปกรณ์ที่ใช้จะเป็นกล่องสำเร็จรูปขนาดเล็กรูปทรงต่างๆ มักมีขนาดกว้าง X ยาว X สูง ประมาณ 10 X 10 X 10 ลูกบาศก์เซนติเมตร ที่พื้นภายในกล่องจะมีแผ่นกรองลักษณะเดียวกับแผ่นกรองในระบบกรองน้ำใต้ทราย จากแผ่นกรองจะมีท่อเล็กๆต่อขึ้นมา 2 ท่อพ้นฝาปิดกล่องขึ้นมา ท่อที่มีปลายขนาดเล็กเป็นท่อสำหรับต่อสายลมจากเครื่องแอร์ปั๊ม บนแผ่นกรองจะมีแผ่นใยสังเคราะห์สำหรับเป็นวัสดุกรองตะกอน โดยมักมีกรวดหรือเศษปะการังใส่ไว้บ้างเพื่อให้กล่องกรองมีน้ำหนัก ด้านบนสุดของกล่องจะมีฝาปิดที่เป็นร่องเล็กๆพอให้ตะกอน เศษอาหาร และสิ่งขับถ่ายของปลาผ่านเข้าไปได้ แต่จะกันปลาไม่ให้เข้าไปในกล่องกรอง เมื่อเปิดแอร์ปั๊มลมจากเครื่องจะผ่านไปออกที่อีกท่อหนึ่งซึ่งอยู่ติดกัน และดันให้น้ำพุ่งขึ้นไปตามท่อเช่นเดียวกัน น้ำในตู้ปลาจะถูกดูดเข้าไปในกล่องกรองหมุนเวียนไปมาตลอดเวลา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.41 รูปภาพเครื่องกรองน้ำ

2.22 ความสำคัญของน้ำต่อการดำรงชีพของปลา

1. เป็นแหล่งออกซิเจนที่ปลาต้องใช้หายใจ ออกซิเจนที่ปลาจะนำไปใช้หายใจได้นั้นจะต้องละลายลงไปในน้ำ สภาพน้ำที่ดีมีการเจือปนของสิ่งต่างๆน้อยจึงจะมีการละลายของ ออกซิเจนได้ดี หรือมีปริมาณของออกซิเจนอยู่สูง จะทำให้ปลาสดชื่นมีสุขภาพดี

2. มีผลต่อการเจริญเติบโต น้ำที่มีคุณภาพเหมาะสมจะทำให้ปลามีการเจริญเติบโตได้ดี สภาพน้ำที่มีการหมักหมมของเศษอาหาร และของเสียจากการขับถ่ายของปลามากเกินไป จะทำให้ปลาแคระแกรนเติบโตช้า ถึงแม้ปลาจะยังมีการกินอาหารดีอยู่ก็ตาม

3. มีผลต่อการกินอาหารของปลา หากสภาพของน้ำมีความไม่เหมาะสมมากขึ้น ปลาจะกินอาหารน้อยลง การว่ายน้ำค่อนข้างเชื่องช้า อ่อนแอและเกิดโรคได้ง่าย

4. มีผลต่อสีสันของปลา น้ำที่มีคุณภาพไม่เหมาะสมจะมีผลทำให้ปลามีสีซีดจางกว่าที่เคยเป็นอยู่

น้ำที่ใช้เลี้ยงปลาสวยงาม

2.23 น้ำที่ใช้เลี้ยงปลาสวยงาม

น้ำประปา เป็นน้ำที่ผู้เลี้ยงปลาสวยงามส่วนใหญ่ใช้เลี้ยงปลากันมากที่สุด โดยส่วนใหญ่จะเป็นผู้เลี้ยงที่อยู่ตามอาคารบ้านเรือน ทำให้สามารถจัดหาได้ง่าย และประการที่สำคัญคือ น้ำประปาจัดว่าเป็นน้ำที่มีความเหมาะสมสำหรับเลี้ยงปลาสวยงามได้เป็นอย่างดี เพราะจากขบวนการของการผลิตน้ำประปา ได้เน้นที่มีความใสสะอาดเพื่อการอุปโภค และบริโภคของมนุษย์ จึงต้องนำเอาน้ำที่มีคุณภาพดีมาผลิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รวมทั้งต้องมีการฆ่าเชื้อโรค จึงทำให้น้ำประปามีความปลอดภัยจากเรื่องโรคพยาธิที่จะมากับน้ำได้ นอกจากนั้นน้ำประปาส่วนใหญ่ยังมีการใช้ปูนขาวช่วยปรับสภาพความเป็นกรดด่างของน้ำ ทำให้น้ำมีความกระด้างและมีความเป็นกรดด่างที่เหมาะสมต่อการเจริญเติบโตของปลา สรุปได้ว่าข้อดีของน้ำประปาในการเลี้ยงปลาสวยงาม คือ ใส ปราศจากโรคพยาธิ และมีคุณสมบัติเหมาะสม

แต่น้ำประปาก็มีปัญหาบางประการในการใช้ คือ น้ำประปาที่พึ่งเปิดออกจากท่อประปามาใหม่ๆนั้น จะมีปัญหาที่สำคัญ 3 ประการ คือ

2.23.1.1 ปริมาณของคลอรีน ซึ่งใช้ในการฆ่าเชื้อโรคในขบวนการผลิต จะยังคงเหลือตกค้างอยู่ในน้ำ ซึ่งมักจะมีความเข้มข้นประมาณ 0.5 - 2.0 มิลลิกรัมต่อลิตร(ppm) ปล่อยปลาลงเลี้ยงทันที หรือเปิดใส่ให้ปลาทันทีทันใด ปริมาณของคลอรีนที่มีอยู่ในน้ำจะมีผลทำให้ปลาตาย ซึ่งเป็นปัญหาที่พบกันอยู่เสมอ ดังนั้นหากต้องการนำน้ำประปาไปใช้เลี้ยงปลาสวยงาม จะต้องทำการกำจัดคลอรีนที่ตกค้างออกก่อน

2.23.1.2 การสะสมก๊าซภายในน้ำ น้ำประปาที่ถูกส่งจากสำนักงานประปาไปตามท่อเพื่อส่งไปยังสถานที่ต่างๆนั้น ในระหว่างที่น้ำไหลไปตามท่อจะเกิดแรงดันทำให้มีก๊าซต่างๆถูกสะสมอยู่ในน้ำเป็นจำนวนมาก จะสังเกตได้ว่าเมื่อเปิดน้ำประปาใส่ภาชนะสักครู่จะมีฟองอากาศเกิดขึ้น โดยเกิดเป็นฟองอากาศเม็ดเล็กๆเกาะอยู่ตามผนังภาชนะ ถ้าเป็นตู้กระจกจะสังเกตได้ชัดเจนมาก แรงดันของก๊าซที่สะสมในน้ำประปานี้จะมีผลต่อปลา โดยจะไปทำให้กล้ามเนื้อของปลาเกิดการขยายตัว กล้ามเนื้อส่วนใดที่อ่อนจะถูกดันให้เกิดการขยายตัวได้ง่าย เช่นที่บริเวณท้องและบริเวณกล้ามเนื้อตาของปลา จึงมักทำให้ปลาเสียการทรงตัวแล้วตาย หรือทำให้ปลาตาปูดโปนออกมา ทำให้ปลามีอาการอักเสบที่ตาแล้วปลามักจะตาบอด วิธีการกำจัดก๊าซที่สะสมในน้ำทำได้ไม่ยากนัก คือในขณะรองน้ำประปาใส่ภาชนะพยายามให้น้ำมีการกระจายตัวออกให้มากที่สุด โดยการเปิดผ่านฝักบัวหรือบีบสายยางฉีดน้ำเหนือภาชนะตลอดเวลา จากนั้นอาจใช้เครื่องให้อากาศปั๊มอากาศเป็นฟองหมุนเวียนอยู่ประมาณ 30 - 60 นาที ก็จะกำจัดก๊าซต่างๆออกได้หมด

2.23.1.3 ความเป็นกรดของน้ำประปา จากขบวนการผลิตน้ำประปาจะมีการใช้สารส้ม เพื่อทำให้เกิดการจับตัวของตะกอนและสารแขวนลอยต่างๆ จากนั้นจึงไปผ่านระบบกรองเพื่อทำให้น้ำใส ผลของการใช้สารส้มจะทำให้น้ำมีคุณสมบัติเป็นกรด ถึงแม้ในระบบการผลิตน้ำประปาจะมีการใช้ปูนขาวเพื่อปรับระดับความเป็นกรดให้มีค่าต่ำลง จนอยู่ในเกณฑ์ที่เหมาะสมต่อผู้บริโภคและไม่น่าจะเป็นอันตรายต่อปลา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.24 คุณสมบัติของน้ำที่เกี่ยวข้องกับการดำรงชีพของปลา

คุณสมบัติของน้ำที่เกี่ยวข้องโดยตรงกับการเจริญเติบโตของปลา หรือการดำรงชีพของปลานั้น จะมีทั้งคุณสมบัติทางด้านกายภาพ (Physical Condition) ทางด้านเคมี (Chemical Condition) และทางด้านชีววิทยา (Biological Condition) ซึ่งมักจะมีความสัมพันธ์กัน หากคุณสมบัติด้านใดด้านหนึ่งไม่เหมาะสม ก็จะส่งผลกระทบไปถึงปลาได้ทันที อาจทำให้ปลาแคระแกรน อ่อนแอป่วยเป็นโรคได้ง่าย หรือไม่สามารถแพร่พันธุ์ได้ ดังนั้นผู้เลี้ยงจำเป็นที่จะต้องระมัดระวังควบคุมคุณสมบัติของน้ำ และพยายามปรับสภาพของน้ำให้เหมาะสมต่อการดำรงชีพของปลาอยู่เสมอ

2.24.1 สีของน้ำ สีของน้ำที่ปรากฏแก่สายตาจำแนกได้เป็น 2 ประเภท คือ

- สีจริง (True Color) เป็นสีที่เกิดจากสารละลายต่างๆที่ละลายในน้ำ ในระยะแรกอาจมองไม่เห็น จนเมื่อมีการสะสมมากขึ้นจึงสังเกตได้ เช่นอาหารปลาบางชนิดจะมีการละลายทำให้น้ำออกเป็นสีเหลือง สีจริงที่เกิดขึ้นจะไม่สามารถแยกออกจากน้ำโดยการกรอง

- สีปรากฏ (Apparent Color) เป็นสีที่สามารถมองเห็นได้ง่าย ส่วนใหญ่เกิดจากตะกอนและสารแขวนลอย รวมทั้งแพลงตอนต่างๆ หรืออาจเกิดจากการสะท้อนแสง

การเลี้ยงปลาสวยงามนั้นน้ำที่มองเห็นในตู้เลี้ยงปลาสวยงามมักจะเป็นสีจริง เนื่องจากการเลี้ยงปลาสวยงามเน้นที่ความใสของน้ำ มีการใช้ระบบกรองน้ำที่ดีและได้รับแสงน้อยจึงไม่มีตะกอนและแพลงตอนเกิดขึ้น สีของน้ำไม่มีผลต่อตัวปลาโดยตรงแต่จะช่วยบ่งบอกได้ถึงความผิดปกติหากความใสของน้ำมีการเปลี่ยนแปลงไป

2.24.2 อุณหภูมิ อุณหภูมิของน้ำมีผลต่อการดำรงชีพของปลาค่อนข้างมาก เพราะปลาเป็นสัตว์เลือดเย็น อุณหภูมิของร่างกายปลาหรือขบวนการเผาผลาญอาหารภายในร่างกายจึงมีการเปลี่ยนแปลงไปตามอุณหภูมิของน้ำ ดังนั้นในช่วงที่มีอุณหภูมิต่ำหรือในฤดูหนาวขบวนการต่างๆในตัวปลาจะลดต่ำลงไปด้วย ซึ่งเท่ากับเป็นการยับยั้งการเจริญเติบโต การกินอาหาร และการแพร่พันธุ์ของปลา ถึงแม้ว่าประเทศไทยจะอยู่ในเขตร้อนมีศักยภาพการเจริญเติบโตของปลาดีกว่าในแถบอื่นตามที่ได้กล่าวมาแล้ว แต่ก็มีช่วงฤดูหนาวซึ่งกินเวลาประมาณ 1 - 2 เดือน ซึ่งก็ต้องถือว่าเป็นระยะเวลาที่มากพอที่จะก่อความเสียหายแก่ปลาที่เลี้ยงได้ ผู้เลี้ยงปลาจะต้องลดปริมาณอาหาร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ที่เคยให้ลง และคอยระวังเรื่องการเกิดโรคระบาด เพราะจากการที่ปลากินอาหารลดลงทำให้สภาพร่างกายมีภูมิต้านทานลดลง จะทำให้ปลาป่วยหรือติดเชื้อต่างๆได้ง่าย จึงพบว่ามักจะเกิดปัญหาโรคระบาดสัตว์น้ำในฤดูหนาวอยู่เสมอ

สำหรับการเลี้ยงปลาสวยงามซึ่งพบปัญหาปลาเกิดโรคระบาดและปลาตายในฤดูหนาวอยู่เสมอ แต่เนื่องจากเป็นการเลี้ยงที่ใช้พื้นที่ไม่มากนัก จึงสามารถที่จะทำการควบคุมอุณหภูมิของน้ำในตู้เลี้ยงปลาได้ โดยใช้เครื่องควบคุมอุณหภูมิ (Heater) ก็สามารถควบคุมให้น้ำมีอุณหภูมิเหมาะสมสำหรับปลาได้ ซึ่งอุณหภูมิที่เหมาะสมสำหรับปลาโดยทั่วไปจะอยู่ระหว่าง 27-32 องศาเซลเซียส การควบคุมอุณหภูมิจะช่วยให้ปลากินอาหารได้ตามปกติ ทำให้ปลาแข็งแรงสุขภาพดี จึงเป็นวิธีการช่วยป้องกันการเกิดโรคระบาดได้อย่างดี

2.24.3 ความขุ่นของน้ำ หมายถึงปริมาณสารแขวนลอยที่อยู่ในน้ำ ความขุ่นของน้ำจะมีผลต่อการบดบังแสง ทำให้ความสมบูรณ์ของบ่อปลาลดลงและยังมีผลอุดตันระบบหายใจ มักทำให้ปลาขนาดเล็กและไข่ปลาตายได้

2.24.4 ความเป็นกรดเป็นด่างของน้ำ มักนิยมเรียกว่า “pH” หมายถึงค่าความเข้มข้นของไฮโดรเจนอิออน (H+)ที่อยู่ในน้ำ ค่า pH ของน้ำจะอยู่ระหว่าง 0-14 โดยมีค่าเป็นกลางที่ pH 7 ที่อุณหภูมิ 25 องศาเซลเซียส สำหรับน้ำธรรมชาติโดยทั่วไปจะมีค่า pH อยู่ระหว่าง 5–9

ระดับค่า pH	ผลต่อการดำรงชีพของปลา
4.0 หรือต่ำกว่า	เป็นอันตรายมักทำให้ปลาตาย
4.1 – 6.0	ปลา จะมีการเจริญเติบโตช้า ผลผลิตต่ำ ระบบสืบพันธุ์ไม่เจริญ
6.5-9	เหมาะสมกับการเลี้ยงปลา
9.1-11.0	ปลาเติบโตช้า
11.1 ขึ้นไป	เป็นอันตราย

ตารางที่ 2.1 ตารางแสดงผลของค่า pH ต่างๆที่ปลาดำรงชีวิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

หลักการทำงานและการออกแบบ

ในการศึกษาโครงงานระบบควบคุมตู้ปลาอัจฉริยะ โดยจะควบคุมการทำงานของอุปกรณ์ต่างๆด้วยหน้าจอแบบสัมผัส และทางโทรศัพท์มือถือผ่านระบบเครือข่ายอินเทอร์เน็ต โดยผู้ศึกษาจะนำเสนอรายละเอียดขั้นตอนการออกแบบระบบ ซึ่งได้นำทฤษฎี และหลักการที่เกี่ยวข้องในบทที่ 2 มาประยุกต์ใช้งานโดยมีขั้นตอนดังนี้

3.1 อุปกรณ์ เครื่องมือ หรือโปรแกรมที่ใช้ในการดำเนินโครงงาน

1. ESP32
2. Nextion 2.8” Basic HMI Display
3. Relay 5V 6 ch
4. Switching power supply 12v
5. LM2596 DC-DC Buck Converter Step Down Module
6. pH Sensor module
7. Temperature Sensor module
8. Tds Sensor Module
9. Micro SD card 32 GB
10. USB Card Reader
11. สายจั๊มเปอร์
12. Protoboard
13. สารละลาย pH buffer 4.01, 6.86 และ 9.18
14. เทอร์โมมิเตอร์
15. เครื่องมือวัด EC/PH
16. พัดลมสำหรับตู้ปลา
17. ฮีทเตอร์สำหรับตู้ปลา
18. ปั๊มดูดสาร
19. ปั๊มกรอง
20. ไฟเส้น 12V
21. Motor 180 องศา 5V
22. ท่อ pvc ขนาด ¾ นิ้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 ขั้นตอนการดำเนินงาน

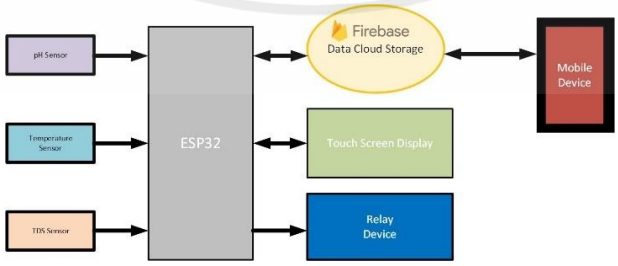
1. ศึกษาเรื่องเกี่ยวกับการควบคุมสภาพน้ำในตู้ปลา
2. ศึกษาอุปกรณ์ที่จะใช้งาน
3. วางจุดประสงค์ และขอบเขตการทำงาน
4. ออกแบบบล็อกไดอะแกรม ผังงาน และผังอุปกรณ์
5. จัดซื้ออุปกรณ์
6. ทำการประกอบอุปกรณ์ ทดสอบการจ่ายไฟ
7. ทำการเขียนโปรแกรมการทำงาน และทดสอบการทำงาน
8. ทำการปรับปรุงแก้ไข ส่วนของวงจร และระบบการทำงาน
9. ทำรายงาน และเตรียมการนำเสนอ

3.3 บล็อกไดอะแกรม(Block Diagram) ผังงาน(Flow Chart) และวงจร(Schematic)

ในโครงงานนี้ในส่วนของการออกแบบนั้น จะเริ่มจากการออกแบบ บล็อกไดอะแกรม (Block diagram) เพื่อแสดงการทำงานภาพรวมในส่วน input การประมวลผล และส่วนของ output จากนั้นจึงออกแบบผังงาน (Flow Chart) การทำงานของโปรแกรมที่จะแสดงผลหรือควบคุมอุปกรณ์ และ ออกแบบวงจร(Schematic) เพื่อจะได้เห็นภาพรวมในการต่ออุปกรณ์ต่างๆเข้าด้วยกัน

3.3.1 บล็อกไดอะแกรม(Block Diagram)

จากรูปที่ 3.1 เป็นการแสดงการทำงานทั้งหมด โดยมี input เป็น Sensor ทั้งหมด 3 ตัว ได้แก่ pH sensor, Temperature sensor และ TDS sensor และสั่งให้ ESP32 แสดงผลทางจอแสดงผลและส่งค่าข้อมูลไปจัดเก็บบนระบบ Cloud ของเว็บไซต์ Netpie และแสดงค่าเซ็นเซอร์ทาง Netpie Freeboard และสั่งงานการเปิดและปิดอุปกรณ์ต่างๆที่ไว้ควบคุมตู้ปลาผ่านสวิชต์ Relay

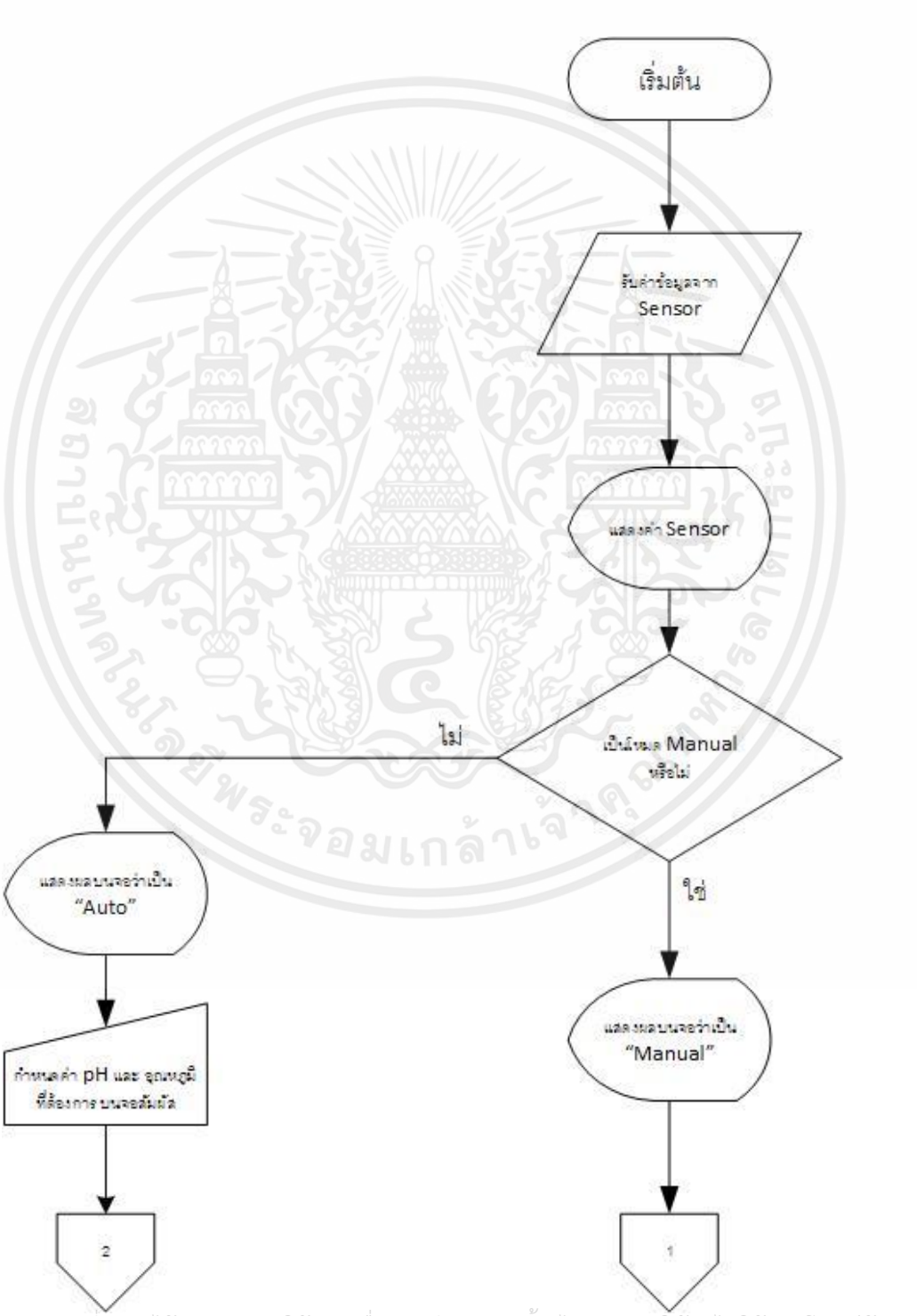


รูปที่ 3.1 รูปภาพแสดงบล็อกไดอะแกรม ของการทำงานต่างๆในส่วน input และ output ของไมโครคอนโทลเลอร์

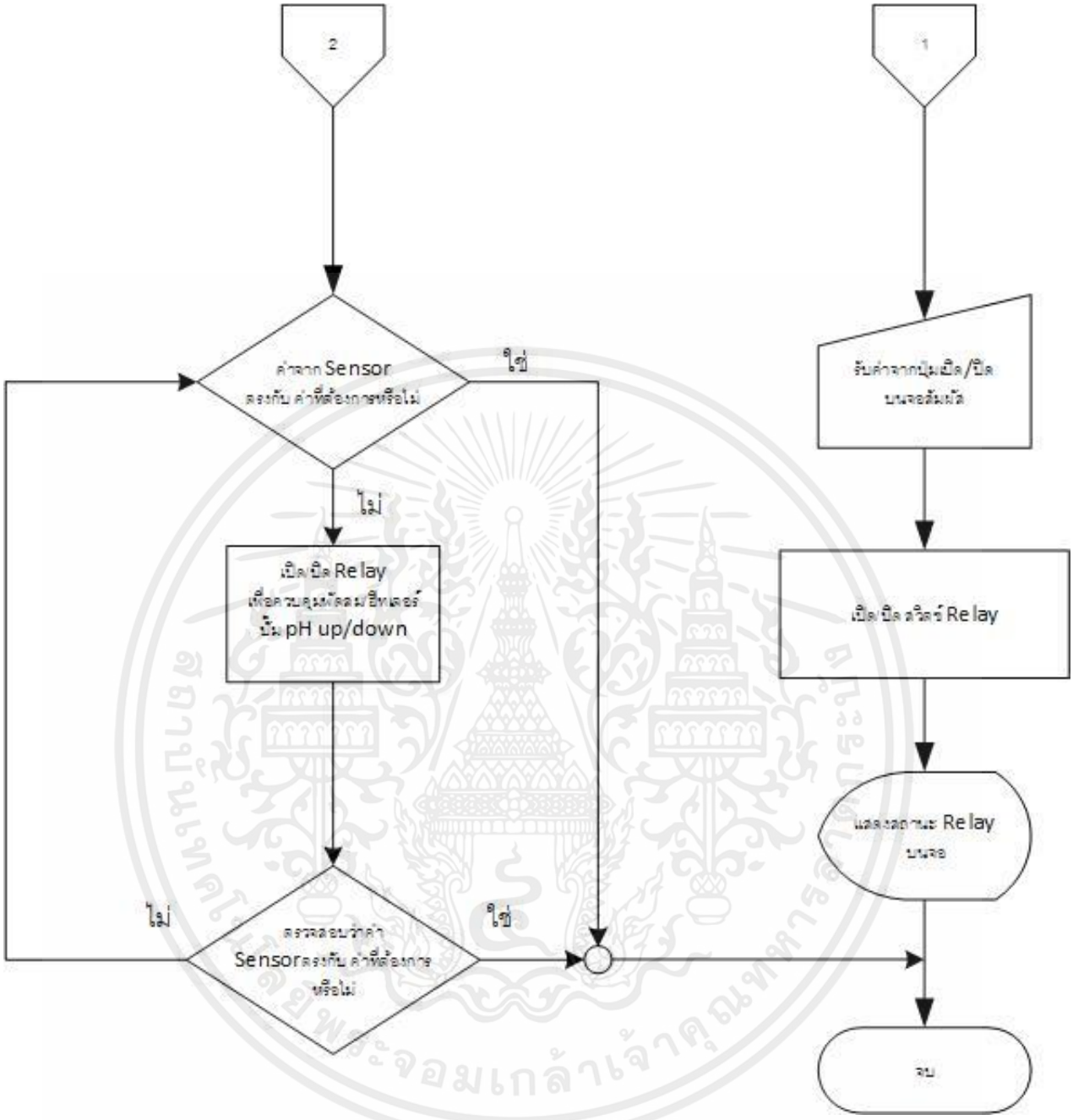
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.2 ผังงาน(Flow Chart)

จากรูปที่ 3.2 เป็นผังงานในส่วนของการควบคุมอุปกรณ์ ซึ่งจะมีโหมดการควบคุมอยู่ 2 โหมด คือ โหมด Manual คือโหมดที่ผู้ใช้จะควบคุมอุปกรณ์ด้วยตัวเองผ่านหน้าจอคอนโทล และโหมด Auto คือผู้ใช้สามารถกำหนดค่าของน้ำที่ต้องการ เช่นต้องการให้น้ำมีค่า pH หรือ Temperature เป็นเท่าไหร่ ระบบก็จะทำการ ประมวลผลและเปิดหรือปิดอุปกรณ์ที่ใช้ควบคุมค่าต่างๆ เช่น เมื่อน้ำมีอุณหภูมิสูงกว่าค่าที่ต้องการ ระบบก็จะทำการเปิดพัดลมระบายความร้อน เป็นต้น



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

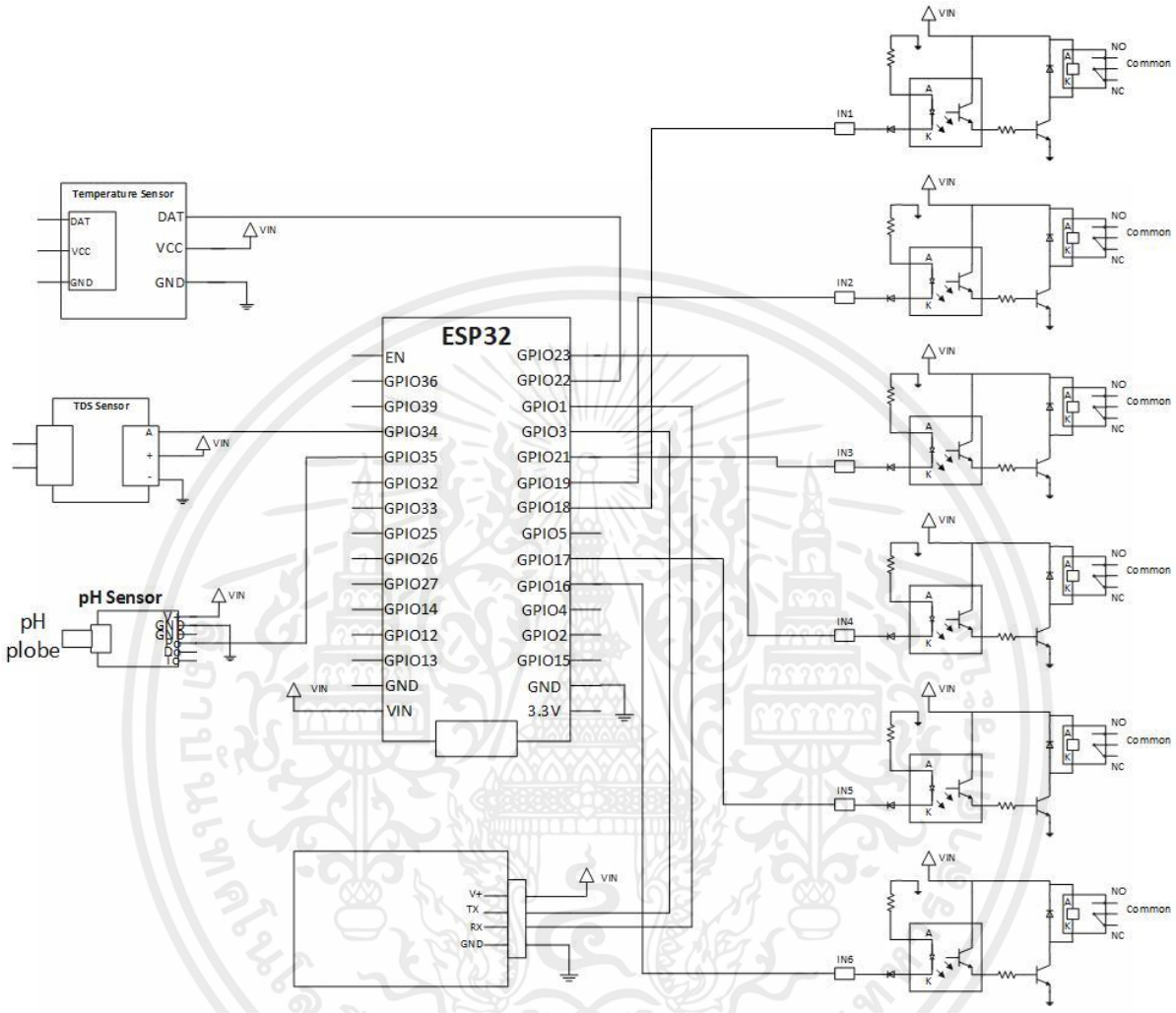


รูปที่ 3.2 รูปภาพผังงานแสดงการทำงานในโหมด Manual และ Auto

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.3 วงจร(Schematic)

รูปที่ 3.3 เป็นวงจรในส่วนการเชื่อมต่อ จาก Sensor และ Relay เข้ากับ ESP32 เพื่อประมวลผลในการแสดงค่าเซนเซอร์ และควบคุม Relay ผ่านทางจอ Nextion Display ที่เป็นจอแบบสัมผัส



รูปที่ 3.3 รูปภาพวงจร แสดงการเชื่อมต่ออุปกรณ์ต่างๆเข้าด้วยกัน

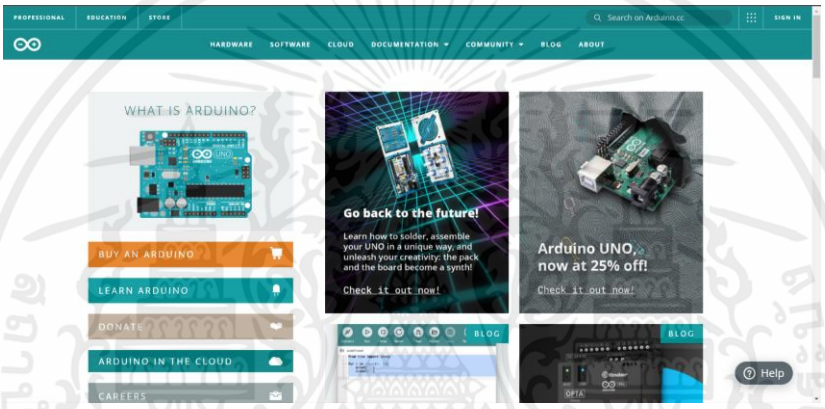
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 การใช้งานโปรแกรม Arduino IDE

Arduino IDE เป็นโปรแกรมที่ใช้สำหรับเขียนโปรแกรมคำสั่งให้กับบอร์ดไมโครคอนโทรลเลอร์ตระกูล Arduino ทุกประเภทรวมถึงบอร์ด ESP32 ที่ผู้จัดทำได้นำมาใช้ในการทำโครงงานนี้ด้วย โดยภาษาของโปรแกรมที่ใช้สั่งการนี้จะใช้ภาษา C/C++ เพื่อเขียนคำสั่งต่างๆ และทำการอัพโหลดลงไปที่เก็บข้อมูลของตัวบอร์ด ด้วยวิธีการใช้งานโปรแกรม Arduino IDE จะมีวิธีการใช้งาน ดังนี้

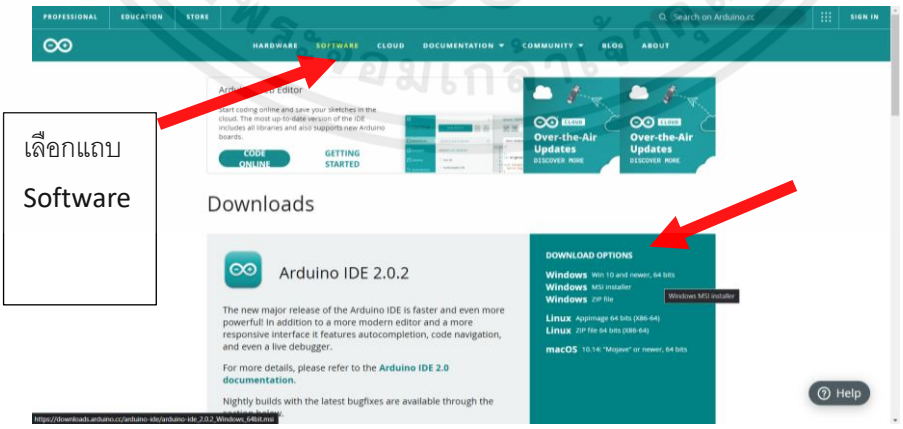
3.4.1 การดาวโหลดและติดตั้งโปรแกรม Arduino IDE

1. เข้าที่หน้าเว็บไซต์ทางการของ Arduino โดยไปที่ https://www.arduino.cc/



รูปที่ 3.4 หน้าเว็บไซต์ของ Arduino

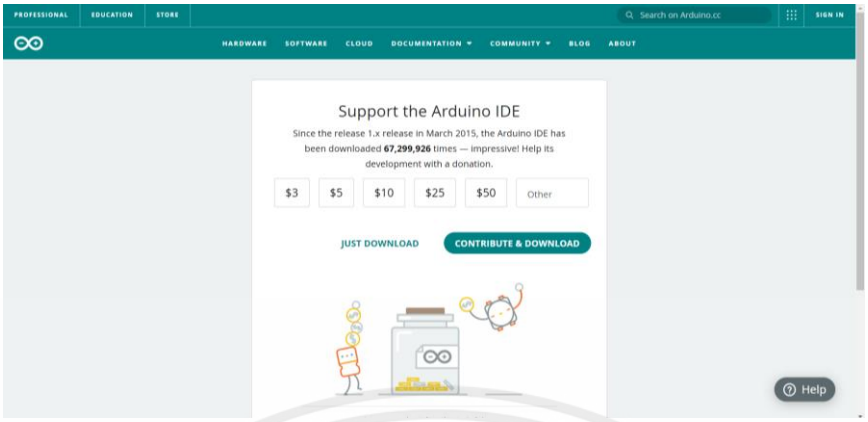
2. เลือกแถบ Software แล้วไปในส่วนของ Download แล้วเลือกในส่วน Download Option เป็นระบบปฏิบัติการ OS ที่ใช้งานอยู่



รูปที่ 3.5 หน้าในของแถบของ Software และปุ่ม Download

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. กด Just Download



รูปที่ 3.6 รูปในหน้า Download

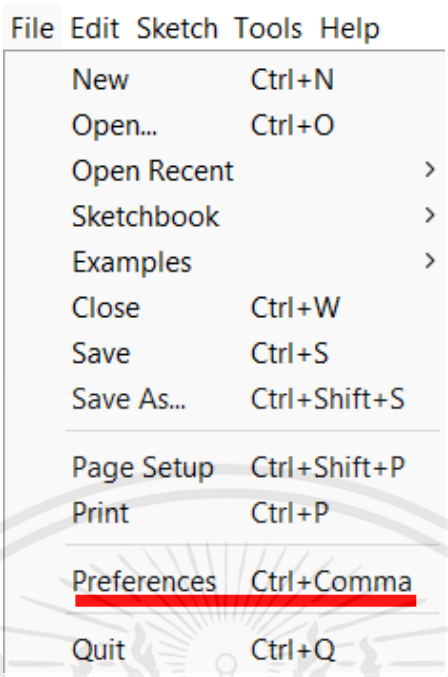
4. เมื่อดาวน์โหลดแล้ว จะได้ไฟล์ Arduino IDE กด Double Click แล้วทำการติดตั้ง เป็นอันเสร็จสิ้นการดาวน์โหลด และติดตั้งโปรแกรม

รูปที่ 3.7 รูป Icon ของตัวติดตั้ง Arduino IDE

3.4.2 การติดตั้งบอร์ด ESP32 บนโปรแกรม Arduino IDE

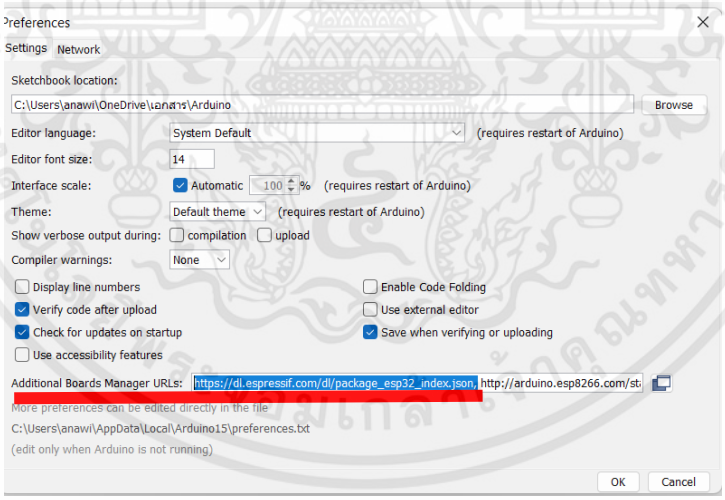
1. ทำการติดตั้งบอร์ด ESP32 ลงในโปรแกรม Arduino IDE โดยกดที่แถบ File แล้วเลือก Preference

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.8 รูปในหน้าแถบ File

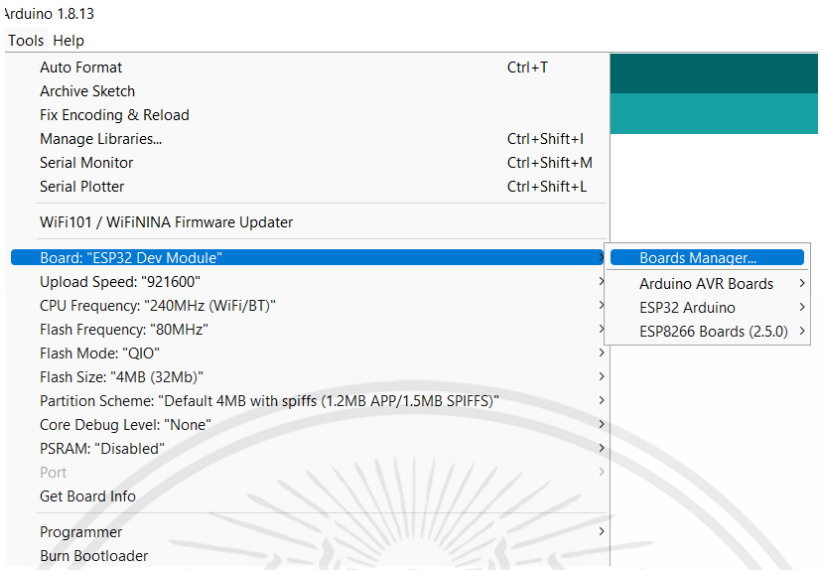
2. ในหน้าต่าง References ให้ใส่ https://dl.espressif.com/dl/package_esp32_index.json ใน Additional Board Manager URLs



รูปที่ 3.9 รูปในหน้าต่าง Preference

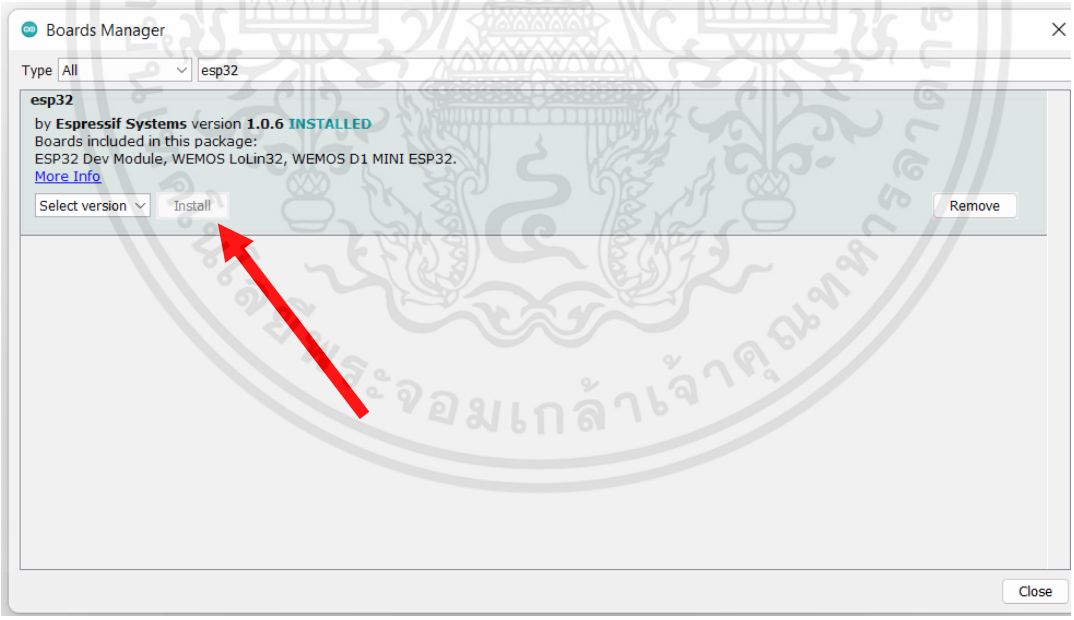
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. จากนั้นกดแถบ Tools แล้วเลือก Board แล้วเลือก Board Manager



รูปที่ 3.10 รูปในหน้าแถบ Tools

4. ในหน้า Boards Manager พิมพ์ในช่องค้นหาว่า “ESP 32” แล้วกด install เสร็จแล้วพร้อมใช้งาน

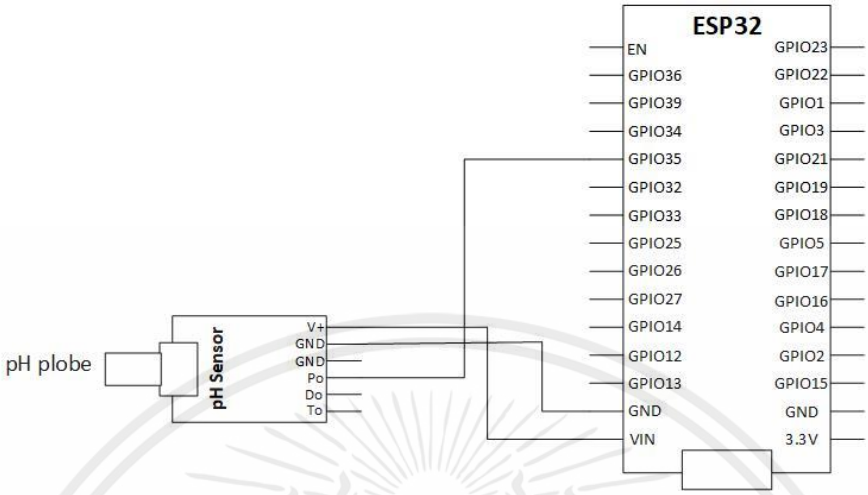


รูปที่ 3.11 รูปในหน้าต่าง Boards Manager

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5 การเชื่อมต่อและใช้งาน pH Sensor กับ ESP32

1. ทำการต่อวงจรโดยเชื่อม pin ของ pH Sensor กับ ESP32 ดังรูปที่ 3.



รูปที่ 3.12 รูปวงจรในส่วนของ pH sensor และ ESP32

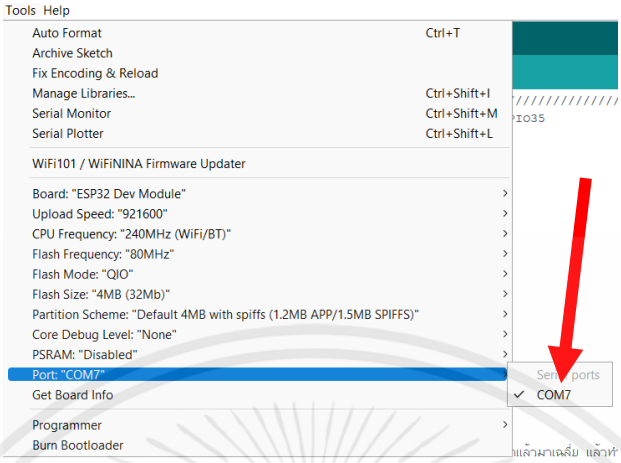
2. ทำการเขียนโปรแกรมรับค่าจาก pH Sensor และแปลงค่าเป็นค่า pH และทำการแสดงค่าผ่านทาง Serial monitor ของ Arduino IDE

```
//////////////////////////////////////////////////     ประกาศตัวแปรต่างๆ     ////////////////////////////////////////////////////
const int analogPhPin = 35;     //   กำหนดขาของ ESp32 ในการรับค่า sensor ในที่นี้ใช้ขา GPIO35
long phTot;
float phAvg;
int x;
float C = 25.85;
float m = -6.80;
//////////////////////////////////////////////////
//////////////////////////////////////////////////
//////////////////////////////////////////////////

void setup() {
   pinMode(35,INPUT);
   Serial.begin(9600);
}

void loop() {
  phTot = 0;
  phAvg = 0;

//////////////////////////////////////////////////  เก็บค่าจาก Sensor ทั้งหมด 10 ค่าแล้วมาเฉลี่ย แล้วทำการแปลงค่าเป็นค่า pH  //////////////
  for(x=0; x<10 ; x++)
    {
        phTot += analogRead(35);
    }
    float phAvg = phTot/10;
    float phVoltage =  phAvg * (3.3 / 4095.0);
    float pHValue = phVoltage*m+C;
//////////////////////////////////////////////////     แสดงค่า pH ทาง Serial monitor     ////////////////////////////////////////
   Serial.print("pH=");
  Serial.println(pHValue);
  delay(500);
}
```

รูปที่ 3.13 รูปภาพแสดงโค้ดการทำงานของ pH sensor

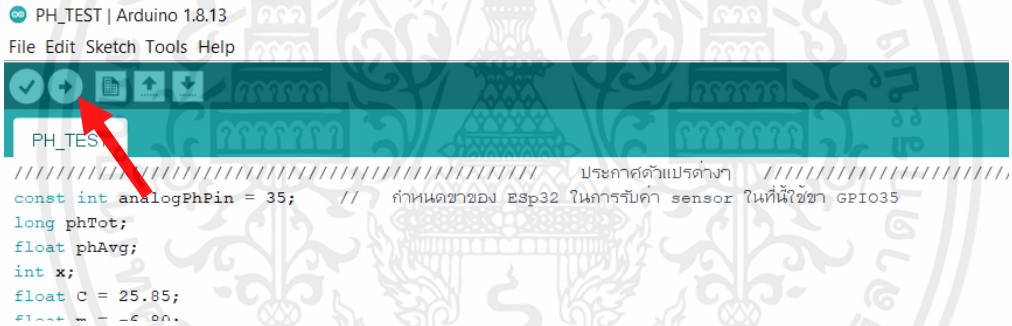
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ทำการอัพโหลดโค้ด โดยไปที่แถบ Tools > Port และเลือก Port ที่เชื่อต่ออยู่ดังรูปที่ 3.14



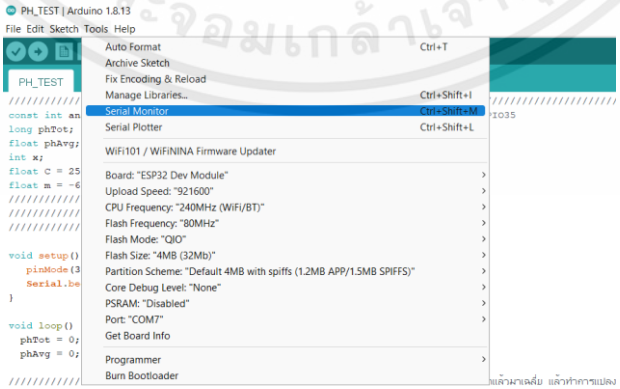
รูปที่ 3.14 รูปแสดงการเลือก Port ที่เชื่อมต่อ

4. ทำการอัพโหลดโปรแกรมลงบอร์ด ESP32 โดยกดปุ่มอัพโหลด ดังรูปที่ 3.15



รูปที่ 3.15 รูปภาพแสดงการอัปโหลดโปรแกรมลงบอร์ด

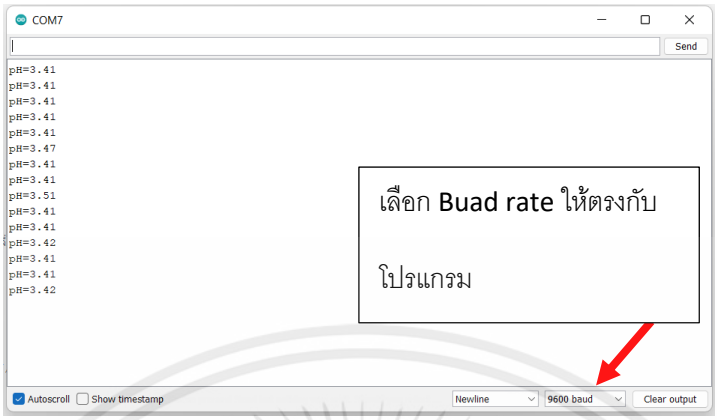
5. ทำการเปิด Serial Monitor โดยไปที่แถบ Tools แล้วเลือก Serial Monitor



รูปที่ 3.16 รูปภาพแสดงการเปิด Serial monitor

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

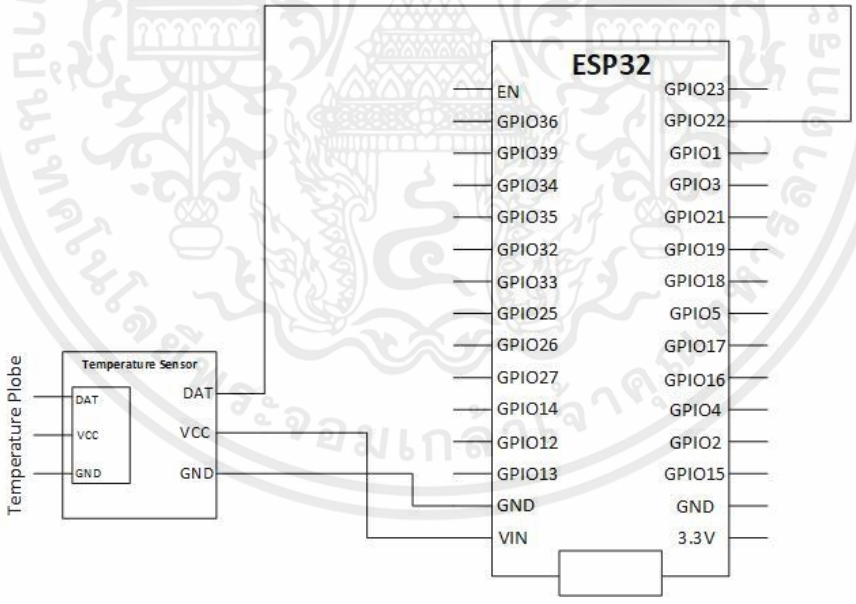
6. ทำการเลือก buad rate ของ Serial Monitor ให้ตรงกับที่กำหนดไว้ในโปรแกรม จากนั้น Serial Monitor จะทำการแสดงค่า PH ออกมาดังรูปที่ 3.17



รูปที่ 3.17 รูปภาพแสดงการเลือก Buad rate และค่า pH ที่แสดงผ่าน Serial Monitor

3.6 การเชื่อมต่อและใช้งาน Temperature Sensor กับ ESP32

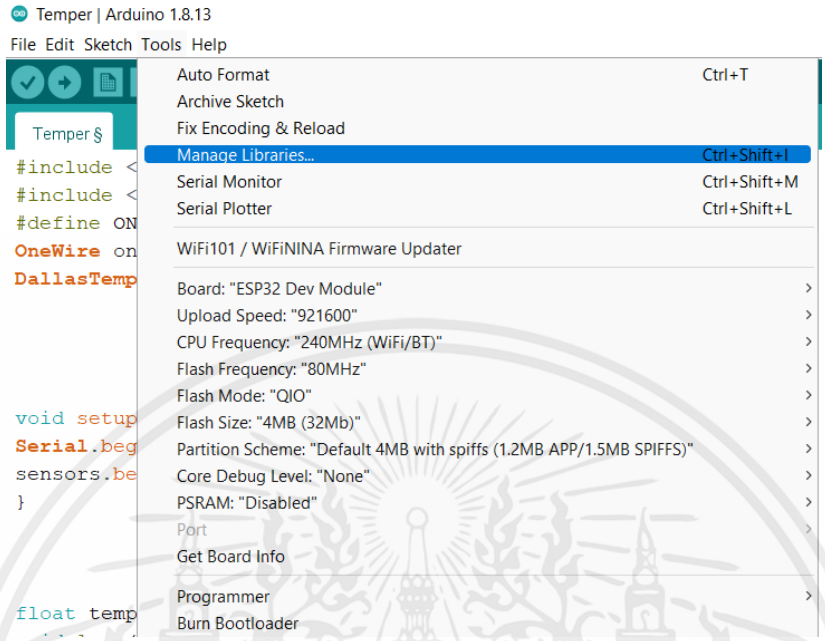
1. ทำการต่อวงจรโดยเชื่อม pin ของ Temperature Sensor กับ ESP32 ดังรูปที่ 3.18



รูปที่ 3.18 รูปวงจรในส่วนของ Temperature sensor และ ESP32

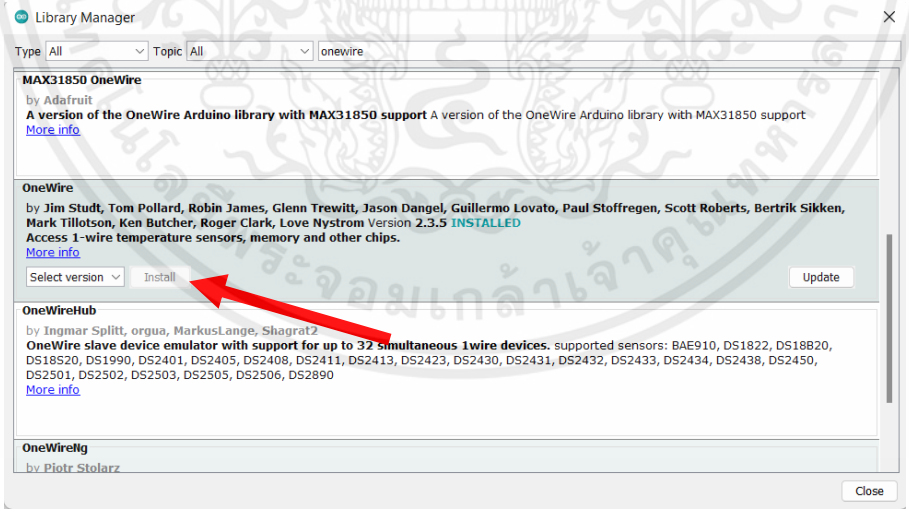
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. ทำการติดตั้งไลบราลี่สำหรับ Temperature sensor โดยไปที่ Tools แล้วเลือก Manage Libraries



รูปที่ 3.19 รูปที่แสดงการเข้าหน้าต่าง Manage libraries

3. ในหน้าต่าง Manage libraries ทำการพิมพ์ในช่องค้นหาว่า “onewire” ซึ่งเป็นไลบรารี่สำหรับใช้งาน sensor แล้วเลือก ไฟล์ที่ชื่อว่า onewire แล้วทำการกด install



รูปที่ 3.20 รูปหน้าต่าง library Manager

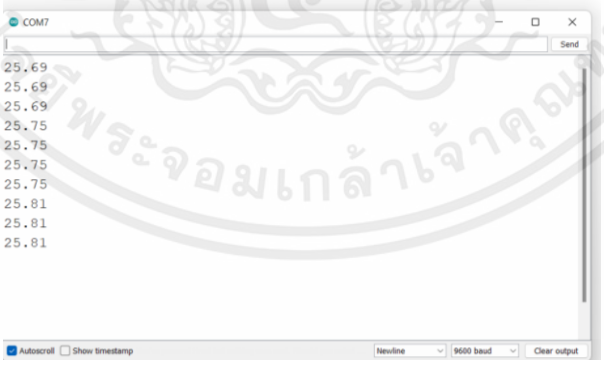
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. ทำการเขียนโปรแกรมรับค่าจาก Temperature Sensor และทำการแสดงค่าผ่านทาง Serial monitor ของ Arduino IDE

```
#include <OneWire.h>                    ////////////////  เรียกใช้ library สำหรับอ่านค่า Sensor /////////////
#include <DallasTemperature.h>          ////////////////////////////////////////////////////////////////////
#define ONE_WIRE_BUS 22                 //  กำหนดขา ESP32 ที่เชื่อมต่อกับ Sensor


OneWire oneWire(ONE_WIRE_BUS);
DallasTemperature sensors(&oneWire);
float temp;                              //ประกาศ์ตัวแปร temp


void setup() {
Serial.begin(9600);                     //กำหนด Buad rate
sensors.begin();
}



void loop (){

sensors.requestTemperatures();          //เรียกใช้ค่าอุณหภูมิ
temp = sensors.getTempCByIndex(0);      //นำค่าอุณหภูมิมาเก็บในค่า temp
Serial.println(temp);                   //แสดงค่า temp ทาง Serial monitor
delay(500);

}
```

รูปที่ 3.21 รูปภาพแสดงโค้ดการทำงานของ Temperature sensor

5. ทำการอัปโหลดโปรแกรมลงบอร์ด ESP32 ดังรูปที่ 3.15 และทำการเปิด Serial monitor ดังรูปที่ 3.16 แล้วจะได้ค่าอุณหภูมิที่แสดงผ่าน Serial monitor ดังรูปที่ 3.22

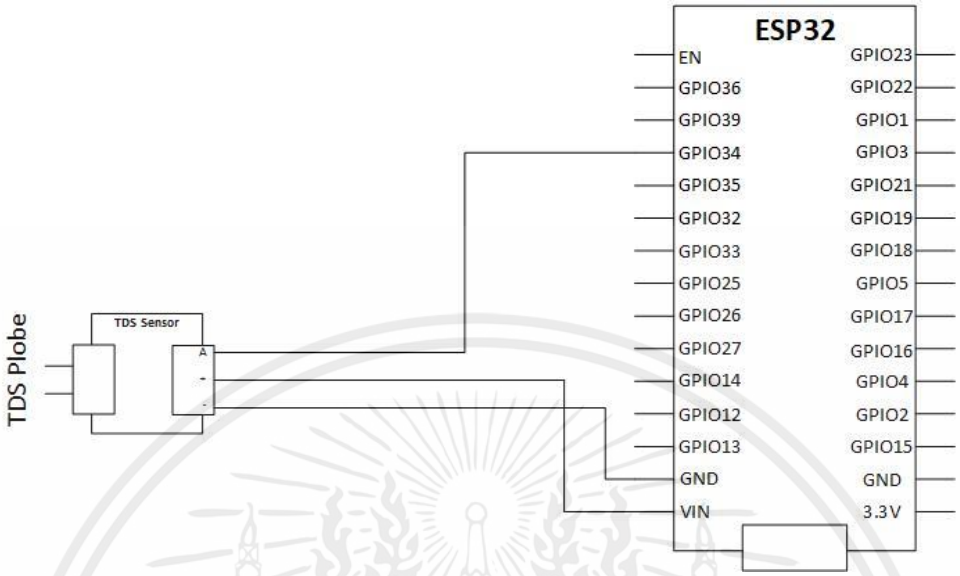


รูปที่ 3.22 รูปแสดงค่าอุณหภูมิที่แสดงผ่าน Serial Monitor

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.7 การเชื่อมต่อและใช้งาน TDS Sensor กับ ESP32

1. ทำการต่อวงจรโดยเชื่อม pin ของ TDS Sensor กับ ESP32 ดังรูปที่ 3.23



รูปที่ 3.23 รูปวงจรในส่วนของ TDS sensor และ ESP32

2.ทำการเขียนโปรแกรมรับค่าจาก TDS Sensor และทำการแสดงค่าผ่านทาง Serial monitor ของ Arduino IDE

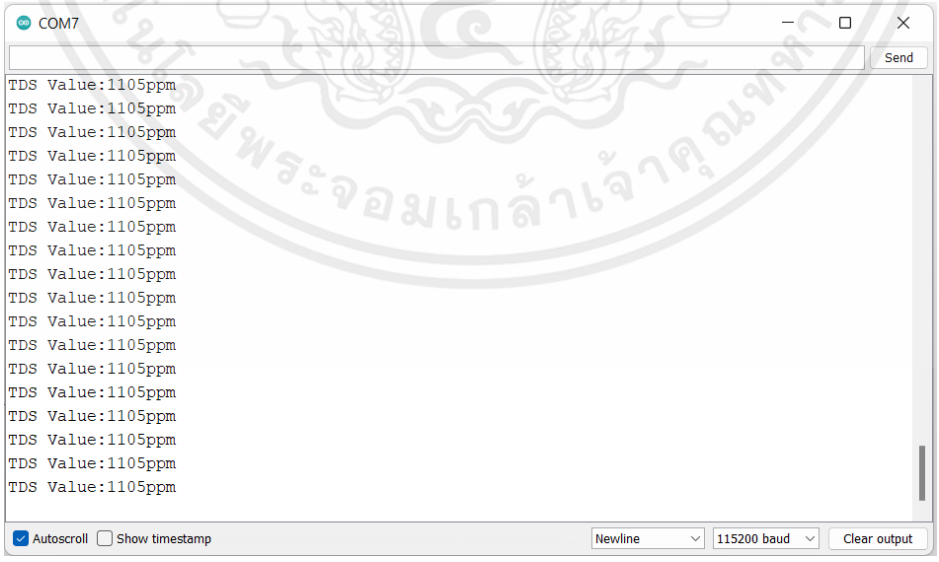
```
#define TdsSensorPin 34         //กำหนดขาของ Esp32 ที่ใช้เชื่อมกับ Sensor
#define VREF 3.3                //Vref ของ esp32
#define SCOUNT  30              //เก็บค่าจากเซ็นเซอร์มา 30 ค่า

int analogBuffer[SCOUNT];       // กำหนดตัวแปร Array
int analogBufferTemp[SCOUNT];
int analogBufferIndex = 0;
int copyIndex = 0;

float averageVoltage = 0;
float tdsValue = 0;
float temperature = 25;

//////////////////////////////////////////////////////////////ประมวลผลหาค่าเฉลี่ยของ Sensor/////////////////////////////////////////////////////////
int getMedianNum(int bArray[], int iFilterLen){
  int bTab[iFilterLen];
  for (byte i = 0; i<iFilterLen; i++)
  bTab[i] = bArray[i];
  int i, j, bTemp;
  for (j = 0; j < iFilterLen - 1; j++) {
    for (i = 0; i < iFilterLen - j - 1; i++) {
      if (bTab[i] > bTab[i + 1]) {
        bTemp = bTab[i];
        bTab[i] = bTab[i + 1];
        bTab[i + 1] = bTemp;
      }
    }
  }
  if ((iFilterLen & 1) > 0){
    bTemp = bTab[(iFilterLen - 1) / 2];
  }
  else {
    bTemp = (bTab[iFilterLen / 2] + bTab[iFilterLen / 2 - 1]) / 2;
  }
  return bTemp;
}
//////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void setup(){
  Serial.begin(115200);
  pinMode(TdsSensorPin,INPUT);
}
//////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
void loop(){
  static unsigned long analogSampleTimepoint = millis();
  if(millis()-analogSampleTimepoint > 40U){                     //อ่านค่า Sensor ทุก 40 millisecond
    analogSampleTimepoint = millis();
    analogBuffer[analogBufferIndex] = analogRead(TdsSensorPin);    //อ่านค่าแล้วเก็บใน Array
    analogBufferIndex++;
    if(analogBufferIndex == SCOUNT){
      analogBufferIndex = 0;
    }
  }

  static unsigned long printTimepoint = millis();
  if(millis()-printTimepoint > 800U){
    printTimepoint = millis();
    for(copyIndex=0; copyIndex<SCOUNT; copyIndex++){
      analogBufferTemp[copyIndex] = analogBuffer[copyIndex];


      averageVoltage = getMedianNum(analogBufferTemp,SCOUNT) * (float)VREF / 4096.0;                    //แปลงค่าเป็นแรงดัน Voltage


      float compensationCoefficient = 1.0+0.02*(temperature-25.0);                                      //เปรียบเทียบกับค่าอุณหภูมิ

      float compensationVoltage=averageVoltage/compensationCoefficient;

      //แปลงค่าแรงดันเป็นค่า TDS
      tdsValue=(133.42*compensationVoltage*compensationVoltage*compensationVoltage - 255.86*compensationVoltage*compensationVoltage + 857.39*compensationVoltage)*0.5;
    }
  }
      Serial.print("TDS Value:");
      Serial.print(tdsValue,0);
      Serial.println("ppm");
      delay(500);
}
```

รูปที่ 3.24 รูปภาพแสดงโค้ดการทำงานของ TDS sensor

3. ทำการอัปโหลดโปรแกรมลงบอร์ด ESP32 ดังรูปที่ 3.15 และทำการเปิด Serial monitor ดังรูปที่ 3.16 แล้วจะได้ค่า TDS ที่แสดงผ่าน Serial monitor ดังรูปที่ 3.25



รูปที่ 3.25 รูปแสดงค่า TDS ที่แสดงผ่าน Serial Monitor

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.8 การใช้งานจอ Nextion 2.8” Basic HMI Display

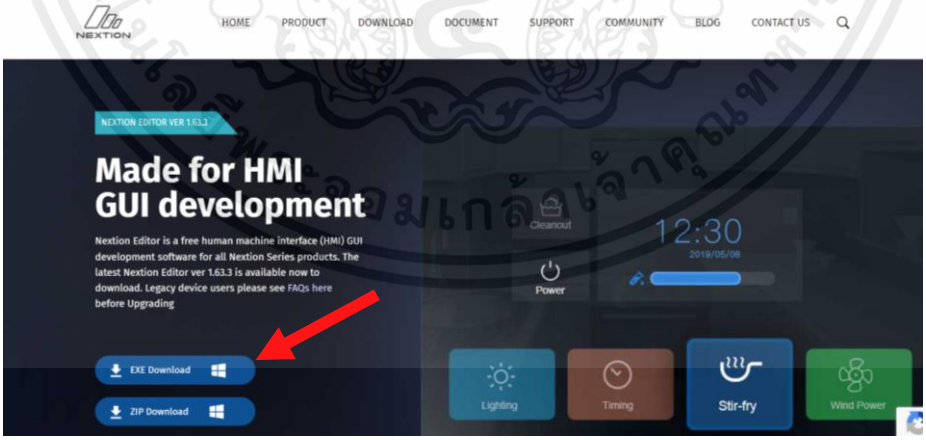
ผู้จัดทำได้ทำการใช้งานจอ Nextion ที่สามารถแสดงค่าจากไมโครคอนโทรลเลอร์ออกมาทางจอภาพ และสามารถควบคุมไมโครคอนโทรลเลอร์ด้วยวิธีการสัมผัสที่จอภาพ และเป็นจอที่มี Interface ที่สวยงามน่าใช้ โดยสามารถปรับแต่งได้โดยใช้ Nextion Editor ซึ่งเป็น Software ที่ใช้กับจอ Nextion โดยตรง ซึ่งสามารถดาวน์โหลดและติดตั้ง และมีวิธีการใช้งานดังนี้

3.8.1 การดาวน์โหลดและติดตั้งโปรแกรม Nextion Editor

1. เข้าไปที่เว็บไซต์ทางการของ Nextion โดยไปที่ https://nextion.tech แล้วกดเลือกที่แถบ Dowload

รูปที่ 3.26 รูปแสดงหน้าเว็บไซต์ของ Nextion

2. กดเลือกดาวน์โหลดตัว .EXE



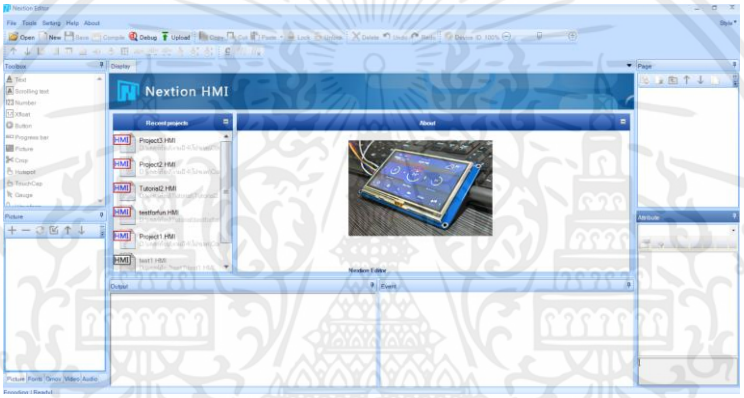
รูปที่ 3.27 รูปแสดงหน้า Download ของ Nextion

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. กด Icon ของ Nextion ที่ Download มาแล้วทำการติดตั้งจนเสร็จ

รูปที่ 3.28 รูป icon ของ Nextion edition ที่ดาวน์โหลดมา

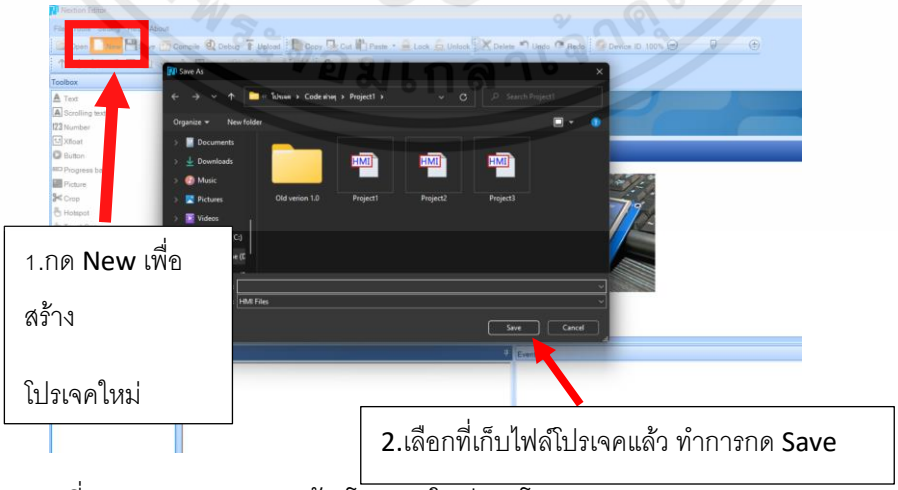
4. เมื่อติดตั้งเสร็จแล้ว จากไปหน้าตาของโปรแกรมดังรูปที่ 3.29 และพร้อมใช้งาน



รูปที่ 3.29 รูปแสดงโปรแกรม Nextion Editor

3.8.2 การเริ่มต้นใช้งาน Nextion Editor

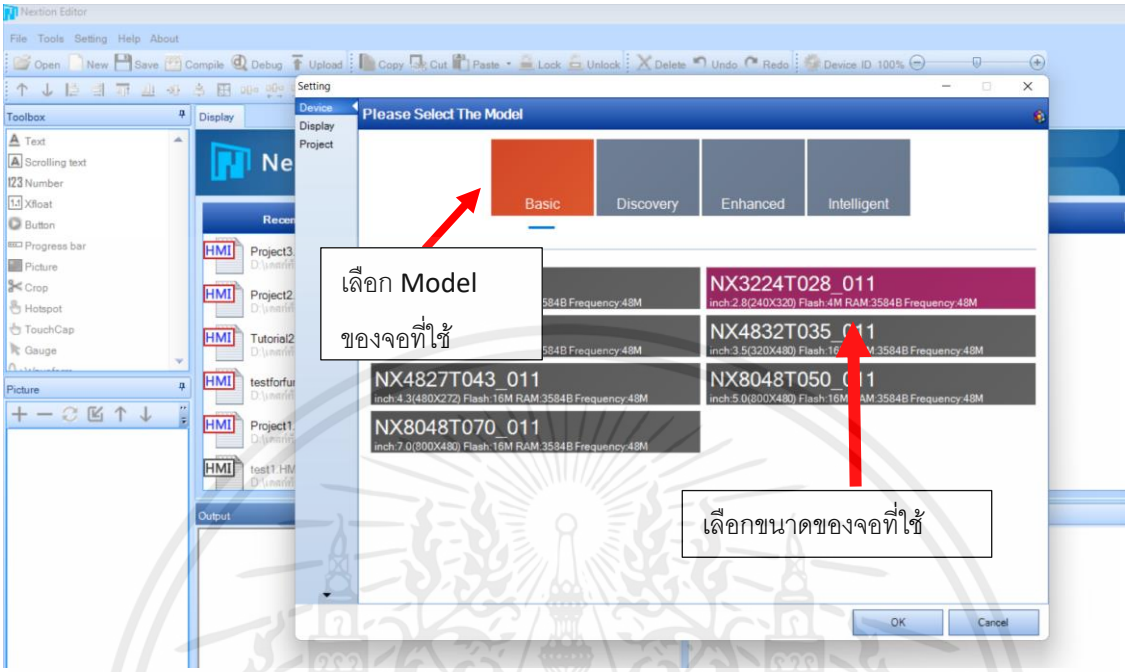
1. ทำการสร้างโปรเจคใหม่โดยกดไปที่ New และทำการ Save project



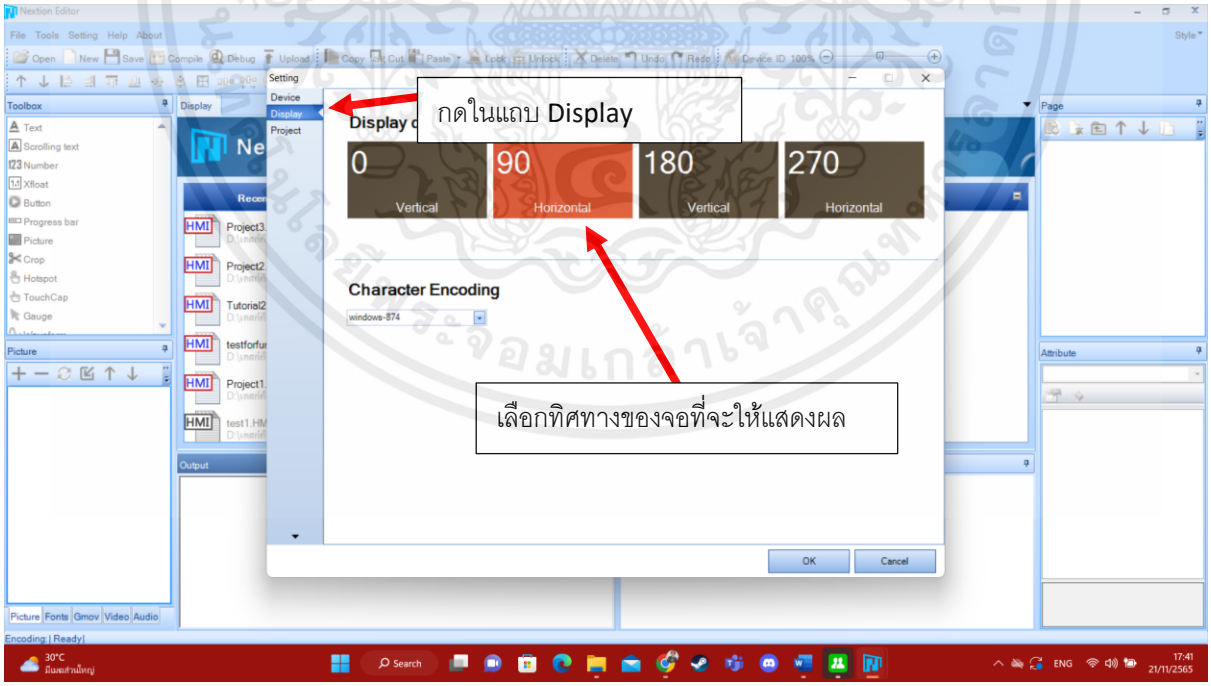
รูปที่ 3.30 รูปแสดงการสร้างโปรเจคใหม่ของโปรแกรม Nextion Editor

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. ทำการเลือก Model และรุ่นของจอ Nextion ที่ใช้งาน ในที่นี้ใช้รุ่น Basic ขนาด 2.8” ดังในรูปที่ 3.31 และทำการเลือกในแถบ Display และเลือกทิศทางของจอที่จะใช้งาน ดังในรูปที่ 3.32



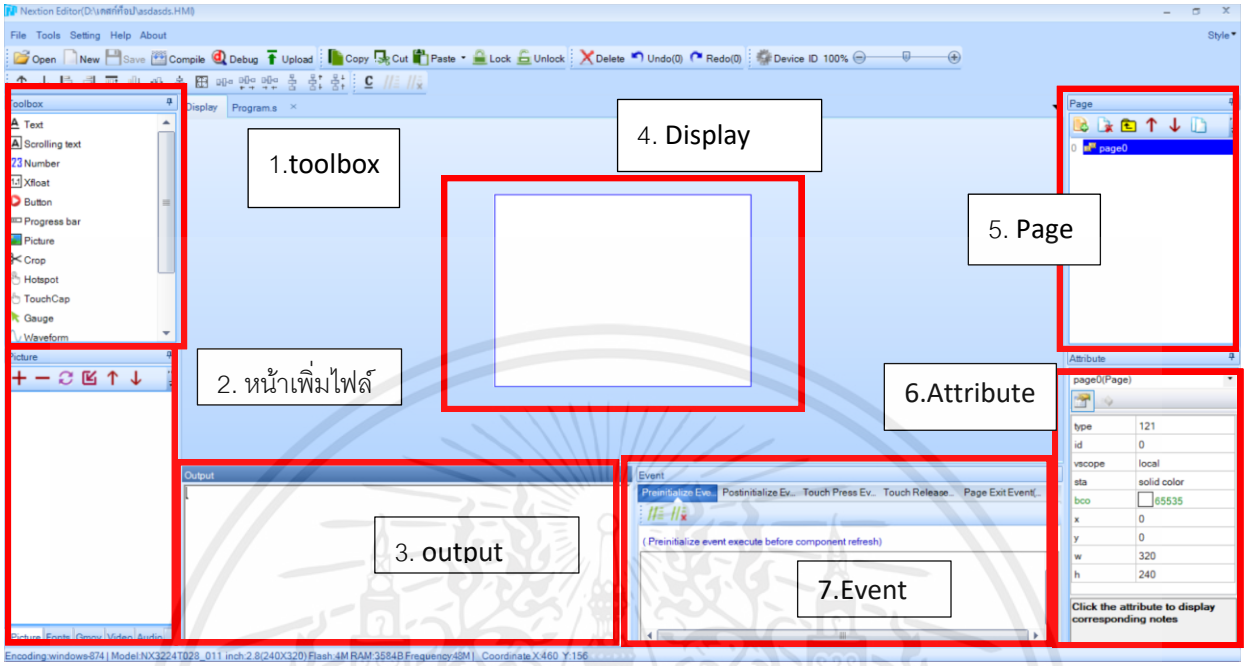
รูปที่ 3.31 รูปแสดงการเลือก Model และ ขนาดของจอที่ใช้งาน



รูปที่ 3.32 รูปแสดงการเลือกทิศทางของจอ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. เมื่อทำการสร้างโปรเจคใหม่ ก็จะสามารถใช้งานการออกแบบจอได้ โดยภายในหน้าต่างโปรแกรมจะมีเครื่องมือและหน้าต่างย่อยต่างๆที่ใช้ในการออกแบบ โดยจะมีเครื่องมือที่ใช้งานโดยหลักๆ ดังรูปที่ 3.33



รูปที่ 3.33 รูปภาพแสดงหน้าต่างของโปรแกรมและหน้าต่างย่อยภายในโปรแกรม

ในรูปที่ 3.33 จะมีหน้าต่างย่อยและเครื่องมือต่างๆ โดยมีรายละเอียด ดังนี้

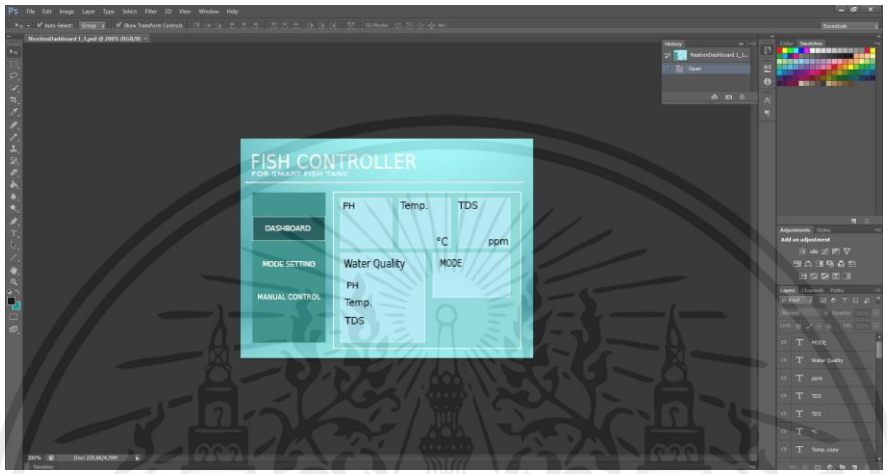
1. หน้าต่าง Toolbox หน้าต่างย่อยนี้จะประกอบด้วยเครื่องมือต่างๆที่สำคัญในการออกแบบหน้าจอ เช่น ปุ่มกด, กล่องข้อความ, แถบสไลด์ เป็นต้น
2. หน้าต่าง เพิ่มไฟล์ โดยหน้าจะเป็นหน้าที่ใช้เพิ่ม ลบ หรือจัดการกับไฟล์รูปภาพ หรือ Font ต่างๆ
3. หน้าต่าง Output จะเป็นหน้าที่แสดงข้อความ Output ขณะที่ทำการ Complile และจะแจ้ง Error ต่างๆในหน้าต่างนี้
4. หน้าต่าง Display ในหน้านี้จะเป็นหน้าที่จำลองจอที่เราจะใช้งานและเป็นหน้าหลักที่จะต้องออกแบบ
5. หน้าต่าง Page เป็นหน้าที่ใช้เพิ่ม ลบ และจัดการกับหน้าต่างๆกรณีที่ต้องการใช้งานหลายๆหน้า
6. หน้าต่าง Attribute เป็นหน้าที่ใช้จัดการคุณสมบัติของเครื่องมือ เช่น ปุ่มกด, กล่องข้อความ เป็นต้น
7. หน้าต่าง Event จะเป็นหน้าต่างที่จะกำหนดคำสั่งให้เครื่องมือนั้นๆทำงานอย่างไร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.9 การแสดงค่า Sensor ทางจอ Nextion Display

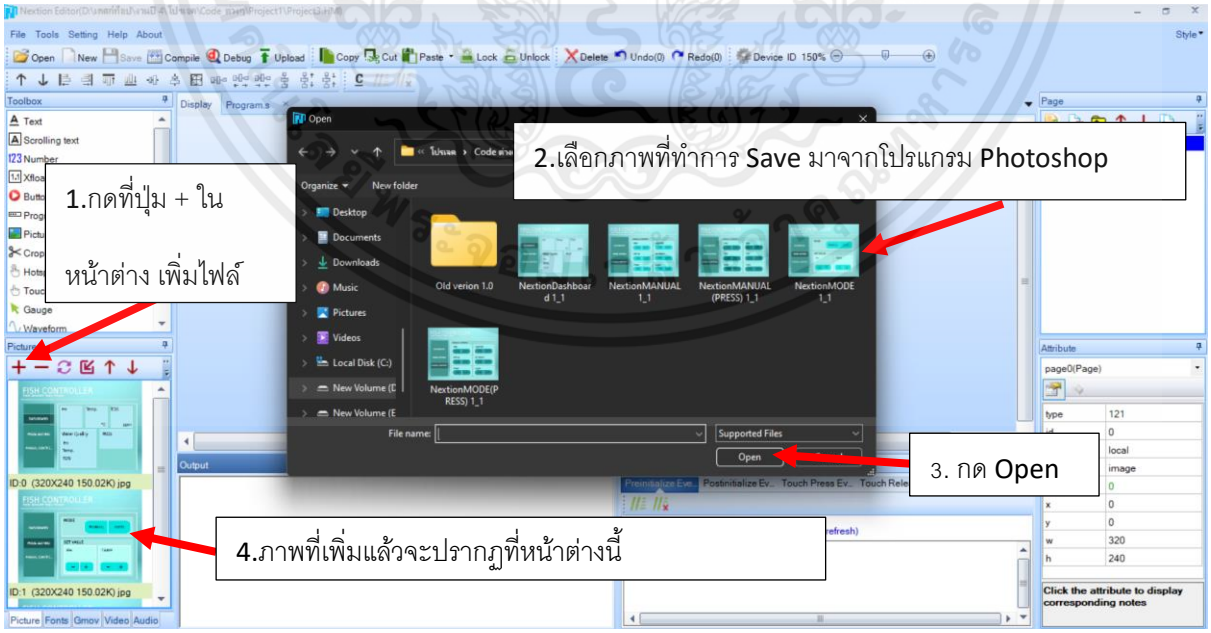
ในโครงงานนี้จะทำการแสดงค่าจาก pH sensor, Temperature sensor, TDS sensor ออกมาแสดงผลทางจอภาพ Nextion Display โดยจะสร้างหน้าที่ใช้แสดงค่าเซ็นเซอร์ทุกตัว ไว้ที่หน้า Dashboard เพื่อความสะดวกในการอ่านค่า โดยมีขั้นตอนดังนี้

1. ทำการออกแบบหน้า Dashboard โดยใช้โปรแกรม Photoshop



รูปที่ 3.34 ภาพแสดงการออกแบบหน้า Dashboard ในโปรแกรม Photoshop

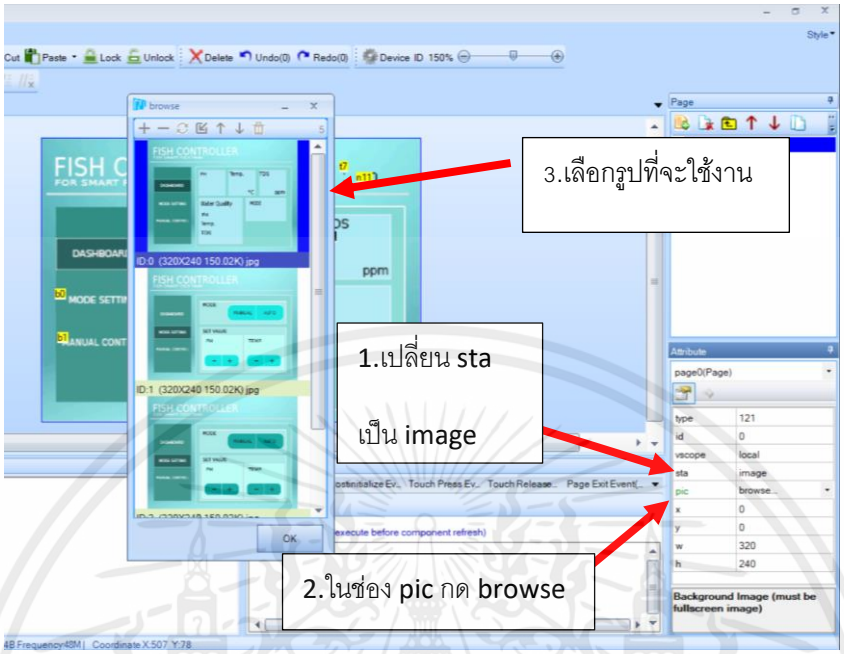
2. ทำการ Save ไฟล์ Dashboard ที่ได้ออกแบบแล้วเป็นรูปภาพและทำการเพิ่มลงไปในโปรแกรม Nextion Editor ในหน้าต่างเพิ่มไฟล์



รูปที่ 3.35 ภาพแสดงการเพิ่มไฟล์รูปภาพในโปรแกรม Nextion Editor

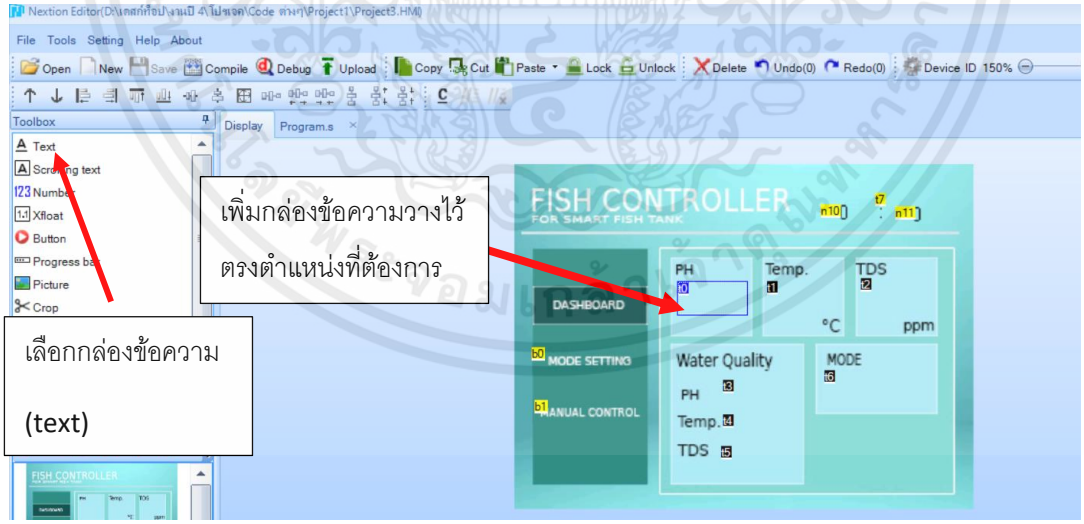
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ในหน้าต่าง Attribute ทำการเปลี่ยน STA เป็น Image และเลือก pic เป็นภาพที่ได้นำเข้ามาดังรูปที่ 3.36



รูปที่ 3.36 ภาพแสดงการเลือกรูปภาพมาใช้งาน

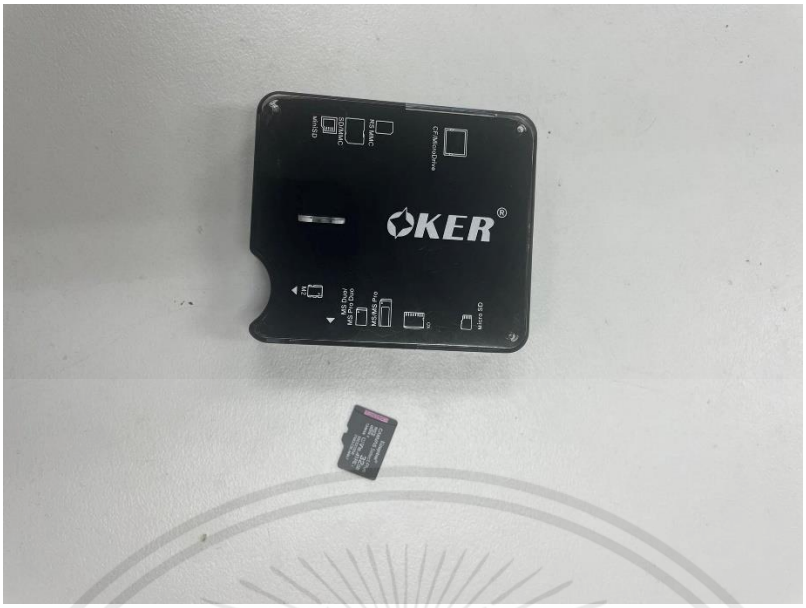
4. ทำการเพิ่มกล่องข้อความเพื่อไว้ใช้แสดงค่าต่างๆ ที่จะส่งมาจากไมโครคอนโทลเลอร์ ตามตำแหน่งที่ต้องการ



รูปที่ 3.37 ภาพที่แสดงการเพิ่มกล่องข้อความ

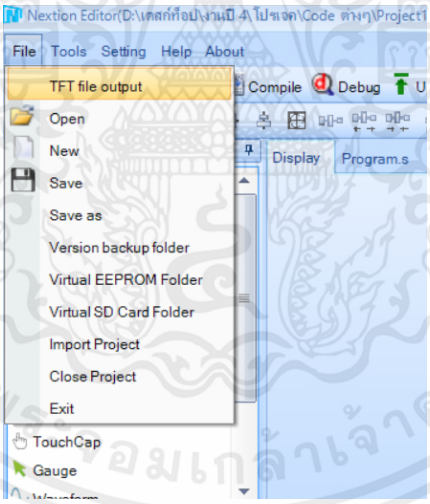
5. เมื่อทำการวางกล่องข้อความตามตำแหน่งที่ต้องการแล้ว จะต้องทำการอัพโหลดไฟล์เพื่อไปใช้งานบนจอ โดยมีวิธีการอัพโหลดด้วยกันหลายวิธี แต่ทางผู้จัดทำได้ใช้วิธีการอัพโหลดโดยใช้ Micro SD card โดยใช้ Card Reader to usb ดังรูปที่ 3.38

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.38 ภาพของ Micro SD card และ Card Reader

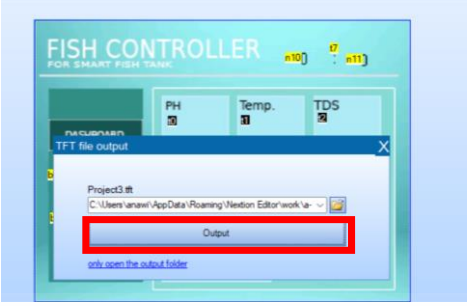
6. ทำการ Save file โดยไปที่ File > TFT File output



รูปที่ 3.39 ภาพแสดงการ Save file ในโปรแกรม Nextion Editor

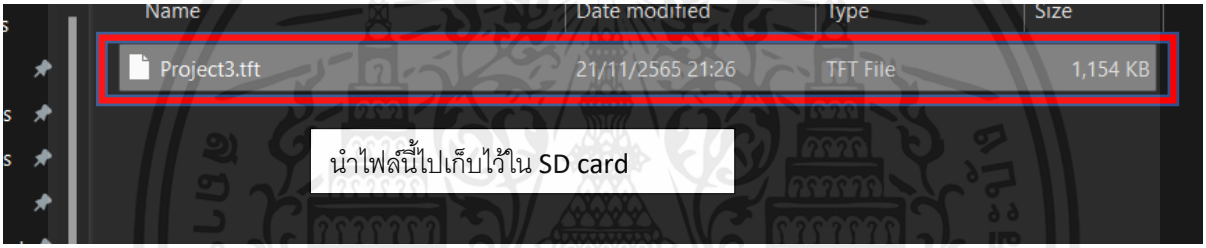
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7. จะขึ้นหน้าต่าง TFT File output โดยทำการกดที่ Output



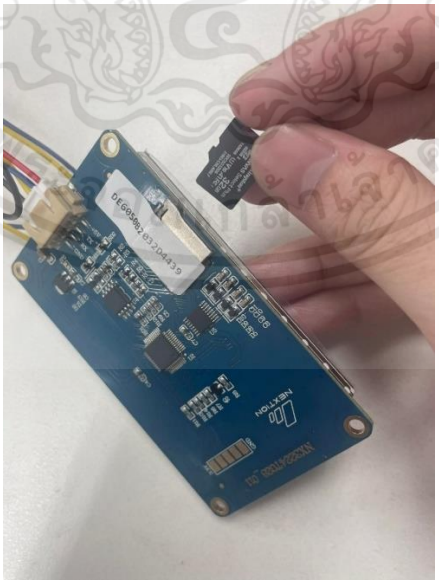
รูปที่ 3.40 หน้าต่าง TFT File output

จะขึ้นหน้าต่างที่เก็บไฟล์ของหน้าจอ ดังรูปที่ 3.41 และทำการลากไฟล์ไปไว้ใน SD card



รูปที่ 3.41 ภาพแสดงไฟล์ของหน้าจอ

8. ทำการใส่ SD card ไปในช่องใส่ SD card ของจอ ดังรูปที่ 3.42



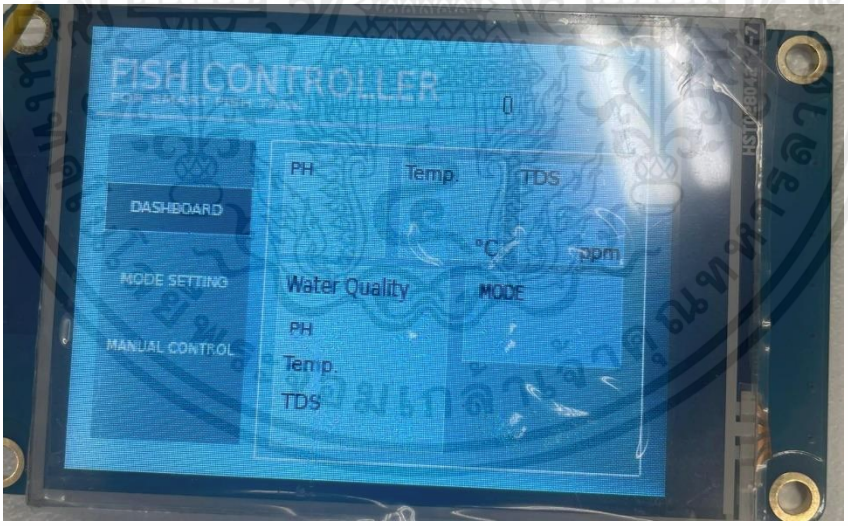
รูปที่ 3.42 ภาพแสดงการใส่ SD card เพื่ออัพโหลดข้อมูลลงจอ Nextion

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทำการจ่ายไฟเลี้ยงให้จอและรอจนอัพโหลดเสร็จ

รูปที่ 3.43 รูปภาพแสดงการอัพโหลดไฟล์ของจอ Nextion

แล้วจึงถอดไฟเลี้ยงจอออก และนำ SD card ออก จากตัวจอ แล้วจ่ายไฟเลี้ยงอีกครั้ง เป็นอันพร้อมใช้งาน



รูปที่ 3.44 รูปภาพของจอ Nextion ที่พร้อมใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

9. ในส่วนของการเขียนโปรแกรมใน Arduino IDE ให้ส่งค่าไป ที่จอ Nextion โดยทำการเขียนดังนี้ โดยในรูปที่ 3.45 จะเป็นการส่งค่าของ pH sensor ซึ่ง Temperature sensor และ TDS ก็ใช้วิธีการเดียวกัน แล้วทำการอัพโหลด

```
phTot = 0;
phAvg = 0;

for(x=0; x<10 ; x++)
  {
      phTot += analogRead(35);
  }
  float phAvg = phTot/10;
  float phVoltage =  phAvg * (3.3 / 4095.0);
  float pHValue = phVoltage*m+C;

Serial.print("t0.txt=\"");      ////////////////ส่งค่าไปที่จอ Nextion โดย ("objectname.ชนิดตัวแปรที่ต้องการส่ง=\")
Serial.print(pHValue);          ////////////////ตามด้วยตัวแปรที่จะส่ง
Serial.print("\"");             ////////////////วรรค
Serial.write(0xff);             ////////////////ส่งค่าให้จอต้องมี3บรรทัดนี้เสมอ
Serial.write(0xff);
Serial.write(0xff);

//////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
```

รูปที่ 3.45 โค้ดในส่วนที่ใช้ส่งค่าไปยังหน้าจอ Nextion

10. เมื่ออัพโหลดโค้ดส่งค่า Sensor แล้วค่าจะขึ้นดังรูปที่ 3.

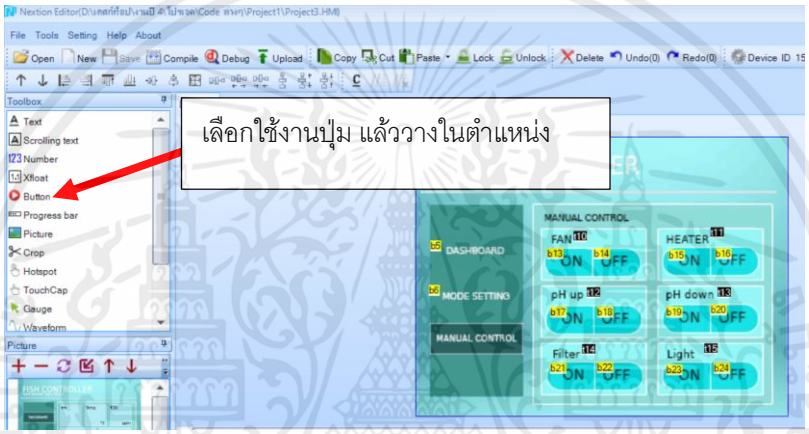
รูปที่ 3.46 ภาพของจอที่แสดงค่าของ Sensor

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.10 การใช้จอ Nextion Display ควบคุมอุปกรณ์

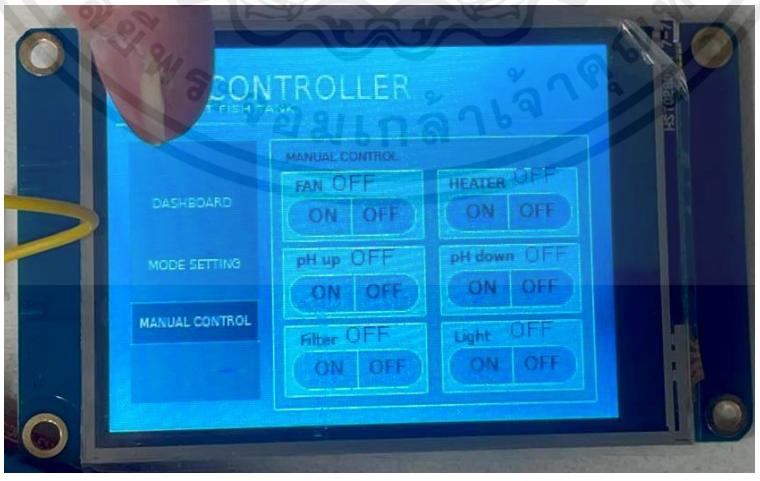
โครงงานนี้ผู้จัดทำได้ทำการใช้จอ Nextion display ที่เป็นจอแบบระบบสัมผัสมาใช้ในการควบคุมอุปกรณ์ โดยใช้การส่งค่าการกดปุ่มบนหน้าจอส่งค่าไปยัง ESP32 เพื่อไปควบคุมอุปกรณ์โดยใช้การเปิดและปิดผ่านรีเลย์ โดยทำการออกแบบหน้าจอที่ใช้เป็นหน้าควบคุม และทำการเพิ่มรูปลงไปในโปรแกรม Nextion เช่นเดียวกับขั้นตอนการแสดงค่าเซ็นเซอร์ที่กล่าวไปในหัวข้อที่ผ่านมา โดยขั้นตอนการใช้งานเครื่องมือปุ่มกดในการสั่งงานอุปกรณ์ มีขั้นตอนดังนี้

1. เมื่อทำการออกแบบหน้าจอมาแล้ว ให้ทำการเพิ่มรูปภาพลงในโปรแกรม Nextion Editor และทำการเพิ่มปุ่มลงในตำแหน่งที่ต้องการ



รูปที่ 3.47 ภาพแสดงการเลือกใช้งานปุ่มกดในโปรแกรม Nextion Editor

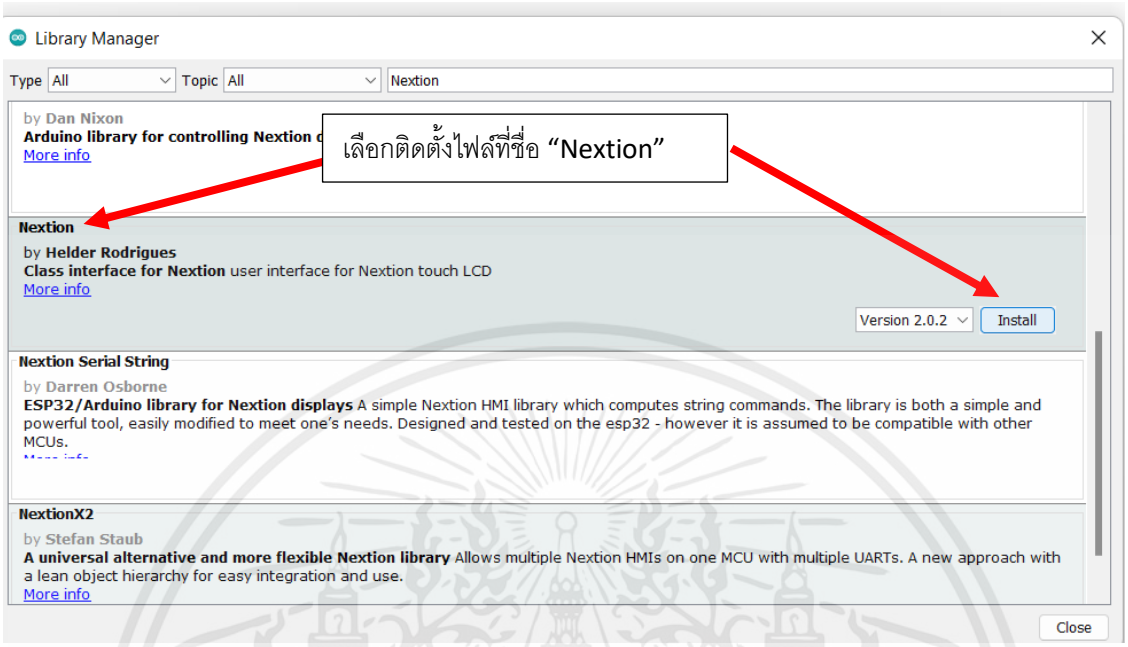
2. ทำการอัพโหลดไฟล์หน้าจอ ลงในจอดังที่กล่าวในหัวข้อที่ 3.9 จะได้หน้าจอที่พร้อมใช้งาน ดังรูปที่ 3.48



รูปที่ 3. 48 รูปภาพแสดงหน้าจอพร้อมปุ่มที่ใช้ควบคุมอุปกรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ในส่วนของการโปรแกรมเพื่อให้ ESP32 เชื่อมต่อกับปุ่มเพื่อให้สามารถสั่งงานรีเลย์ได้ ในเริ่มแรกให้ทำการติดตั้ง library ของ Nextion ในโปรแกรม Arduino IDE



รูปที่ 3.49 รูปภาพแสดงการติดตั้ง library ของ Nextion

4. ทำการเขียนประกาศตัวแปรของปุ่มที่ใช้ใน Nextion Editor ดังรูปที่ 3.50

```
NexButton b7 = NexButton(1, 3, "b7");
NexButton b8 = NexButton(1, 4, "b8");
NexButton b9 = NexButton(1, 5, "b9");
NexButton b10 = NexButton(1, 6, "b10");
NexButton b11 = NexButton(1, 7, "b11");
NexButton b12 = NexButton(1, 8, "b12");
NexButton b13 = NexButton(2, 3, "b13");
NexButton b14 = NexButton(2, 4, "b14");
NexButton b15 = NexButton(2, 5, "b15");
NexButton b16 = NexButton(2, 6, "b16");
NexButton b17 = NexButton(2, 7, "b17");
NexButton b18 = NexButton(2, 8, "b18");
NexButton b19 = NexButton(2, 9, "b19");
NexButton b20 = NexButton(2, 10, "b20");
NexButton b21 = NexButton(2, 11, "b21");
NexButton b22 = NexButton(2, 12, "b22");
NexButton b23 = NexButton(2, 13, "b23");
NexButton b24 = NexButton(2, 14, "b24");
NexButton b25 = NexButton(2, 21, "b25");
NexButton b26 = NexButton(2, 22, "b26");

NexTouch *nex_listen_list[]={&b7,&b8,&b9,&b10,&b11,&b12,&b13,&b14,&b15,&b16,&b17,&b18,&b19,&b20,&b21,&b22,&b23,&b24,&b25,&b26,NULL};
```

รูปที่ 3.50 รูปภาพแสดงการประกาศตัวแปรของปุ่มที่ใช้ใน Nextion Editor

5. กำหนดหน้าที่ให้กับปุ่มนั้นๆ โดยการสร้างฟังก์ชั่นของแต่ละปุ่ม ดังรูปที่ 3.51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
////////////////////////////////Light/////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
void b23PushCallback(void *ptr)                    //////////////สร้างฟังก์ชันการทำงานของปุ่ม 23(pushcallback คือทำงานเมื่อกด popcallback คือทำงานเมื่อปล่อยปุ่ม//////////////
{
 digitalWrite(17,HIGH);                            ////////////// สั่งเปิด Relay ที่ขา 17 ///////////////////////
 Serial.print("t15.txt=");
 Serial.print("\"");
 Serial.print("ON");
 Serial.print("\"");
 Serial.write(0xff);
 Serial.write(0xff);
 Serial.write(0xff);
}
void b24PushCallback(void *ptr)                    //////////////สร้างฟังก์ชันการทำงานของปุ่ม 23(pushcallback คือทำงานเมื่อกด popcallback คือทำงานเมื่อปล่อยปุ่ม//////////////
{
 digitalWrite(17,LOW);                             //////////////สั่งเปิด Relay ที่ขา 17 ///////////////////////
 Serial.print("t15.txt=");
 Serial.print("\"");
 Serial.print("OFF");
 Serial.print("\"");
 Serial.write(0xff);
 Serial.write(0xff);
 Serial.write(0xff);
}
////////////////////////////////////////////////////////////////////////////////////////////servo/////////////////////////////////////////
```

รูปที่ 3.51 รูปภาพของโค้ดที่สร้างฟังก์ชั่นของปุ่มต่างๆ

6. ในส่วนของ void setup() ทำการเรียกฟังก์ชั่นของปุ่มที่ใช้งานในส่วนนี้ ดังรูปที่ 3.52

```
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
////////////////////////////////////////////////////////////// เรียกใช้ฟังก์ชันการทำงานของปุ่มในส่วนของ   set up  //////////////////////////////////////////////////////////

  b7.attachPush(b7PushCallback,&b7);
  b8.attachPush(b8PushCallback,&b8);
  b9.attachPush(b9PushCallback,&b9);
  b10.attachPush(b10PushCallback,&b10);
  b11.attachPush(b11PushCallback,&b11);
  b12.attachPush(b12PushCallback,&b12);
  b13.attachPush(b13PushCallback,&b13);
  b14.attachPush(b14PushCallback,&b14);
  b15.attachPush(b15PushCallback,&b15);
  b16.attachPush(b16PushCallback,&b16);
  b17.attachPush(b17PushCallback,&b17);
  b18.attachPush(b18PushCallback,&b18);
  b19.attachPush(b19PushCallback,&b19);
  b20.attachPush(b20PushCallback,&b20);
  b21.attachPush(b21PushCallback,&b21);
  b22.attachPush(b22PushCallback,&b22);
  b23.attachPush(b23PushCallback,&b23);
  b24.attachPush(b24PushCallback,&b24);
  b25.attachPush(b25PushCallback,&b25);
  b26.attachPush(b26PushCallback,&b26);


////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
```

รูปที่ 3.52 ภาพในส่วนของโค้ดที่ทำการเรียกฟังก์ชั่นของปุ่มที่ใช้งาน

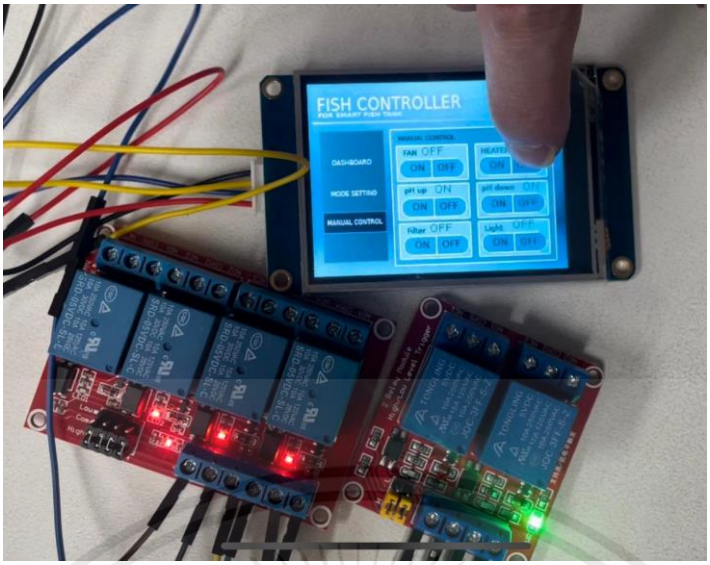
7. ในส่วนของ void loop ทำการรับค่าจากปุ่มทุกครั้งที่มีการกด ดังรูปที่ 3.53

```
void loop()
{
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

  nexLoop(nex_listen_list);                 //////// รับค่าจากปุ่มทุกครั้งที่มีการกด

////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
```

รูปที่ 3.53 ภาพในส่วนของการรับค่าจากปุ่ม

8. ทำการอัพโค้ดและทดสอบการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

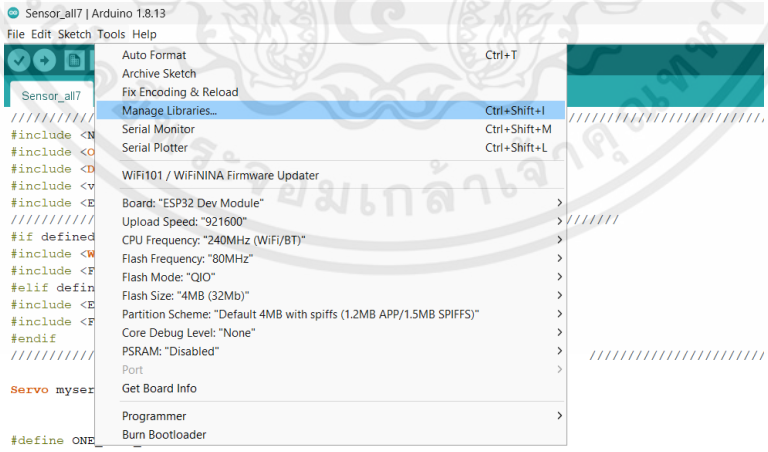


รูปที่ 3.54 ภาพการทดสอบการปิดและเปิดรีเลย์ผ่านหน้าจอสัมผัส

3.11 การส่งค่าไปยัง Firebase Realtime Database

ในโครงงานนี้นอกจากการนำค่าจาก Sensor มาแสดงผลบนหน้าจอ LCD แล้วนั้นทางผู้จัดทำยังได้นำค่าจาก Sensor ที่ได้ส่งขึ้นไปไว้ยัง Cloud database เพื่อที่จะนำค่าเหล่านี้ไปใช้แสดงผลผ่านระบบอินเทอร์เน็ต เพื่อให้ผู้ใช้งานสามารถที่จะตรวจสอบค่าเหล่านี้ได้ทุกที่ ทุกเวลาผ่านระบบอินเทอร์เน็ต โดยการใช้แสดงผลผ่านแอพพลิเคชันในมือถือ โดยจะมีวิธีการดำเนินการดังนี้

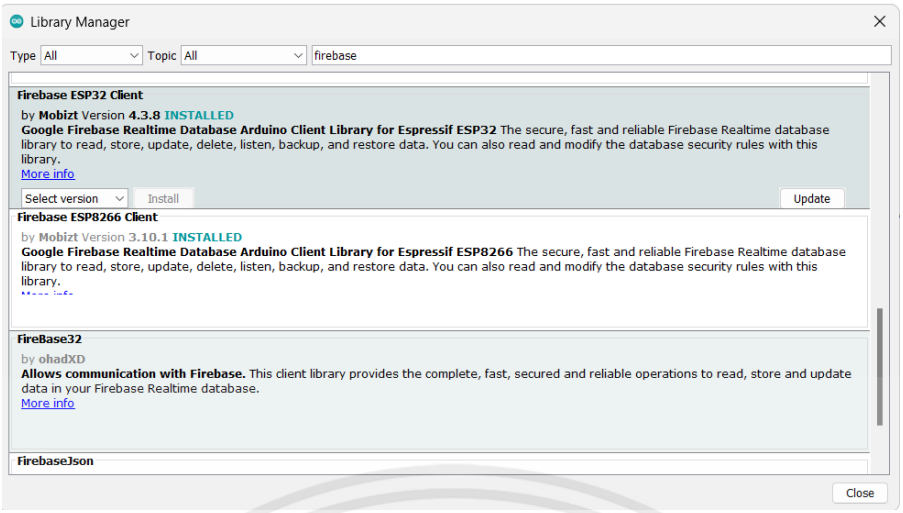
1. ในการเขียนโปรแกรมส่งข้อมูลไปยัง Firebase นั้นจะต้องทำการติดตั้ง Library ของ Firebase โดยไปที่ Tool แล้วเลือก Manage libraries



รูปที่ 3.55 รูปภาพแสดงการเข้าเมนู Manage Libraries

2.ทำการติดตั้ง Libraries ที่ชื่อว่า Firebase ESP32 Client

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.56 รูปภาพแสดงการติดตั้ง library firebase ESP32 Client

3. ทำการ include library ของ Firebase ในส่วนของการเขียนโปรแกรม ขึ้นมา

```
#if defined(ESP32)
#include <WiFi.h>
#include <FirebaseESP32.h>
#elif defined(ESP8266)
#include <ESP8266WiFi.h>
#include <FirebaseESP8266.h>
#endif
```

รูปที่ 3.57 รูปภาพแสดงการ Include library ของ firebase

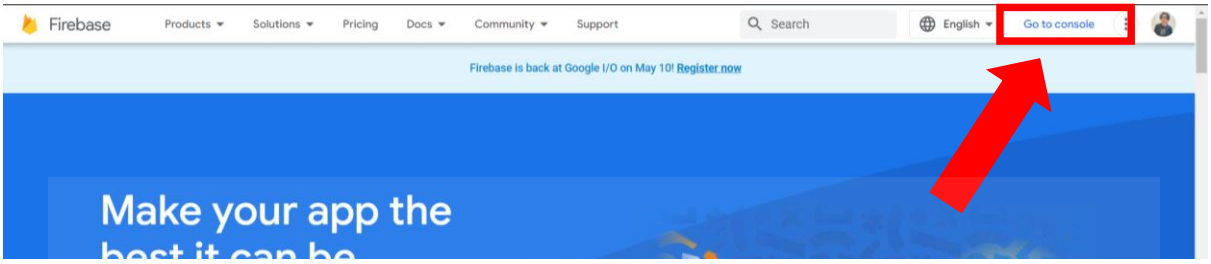
4. ทำการกำหนดค่าเริ่มต้นต่างๆของส่วน Internet โดยการใส่ SSID Wifi, WIFI Password และ ตัวแปรต่างๆของการทำงาน Firebase เช่น API Key และ DATABASE url เพื่อทำการเชื่อมต่อกับ Firebase โดยวิธีการรับค่าเหล่านี้ จะกล่าวในขั้นตอนถัดไป

```
//////////////////////////////////// INTERNET ////////////////////////////////////////////

#include <addons/TokenHelper.h>
#include <addons/RTDBHelper.h>


#define WIFI_SSID "joe"
#define WIFI_PASSWORD "joe55555"



#define API_KEY "WrBpaBAeQW6Lwgkl5DR2qA8WdbVjjFO22JBdt30w"
#define DATABASE_URL "https://project-384473881371097354-default-rtdb.asia-southeast1.firebasedatabase.app/" //<databaseName>.firebaseio.com or <databaseName>.<region>.firebasedatabase.app


FirebaseAuth auth;
FirebaseConfig config;


String main="";
```

รูปที่ 3.58 รูปภาพแสดงการกำหนดค่าเริ่มต้นของ Internet และ Firebase

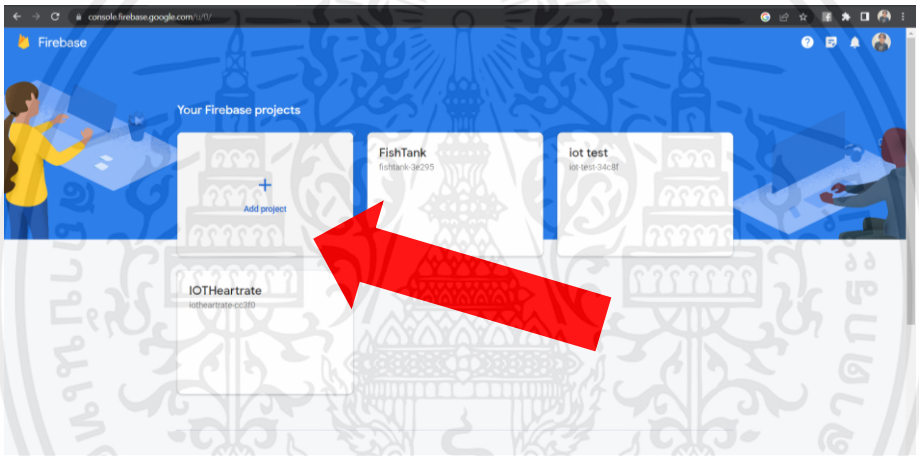
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. ในส่วนของ Firebase ให้ทำการไปที่เว็บไซต์ https://firebase.google.com ทำการ Sign up และ Sign in ให้เรียบร้อยแล้วไปที่ Go to console



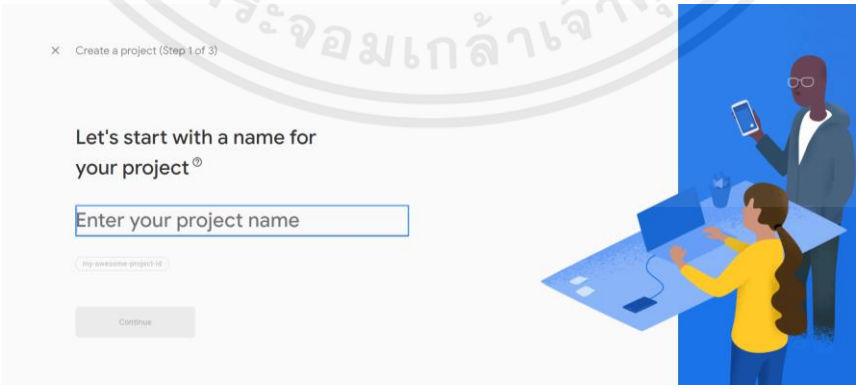
รูปที่ 3.59 รูปภาพแสดงตำแหน่งของปุ่ม Go to console

6. เมื่อเข้ามาแล้วในกรณีที่ยังไม่ได้สร้างโปรเจคสามารถสร้างได้ โดยกดที่ Add project ดังรูป



รูปที่ 3.60 รูปภาพแสดงหน้าสร้าง Project

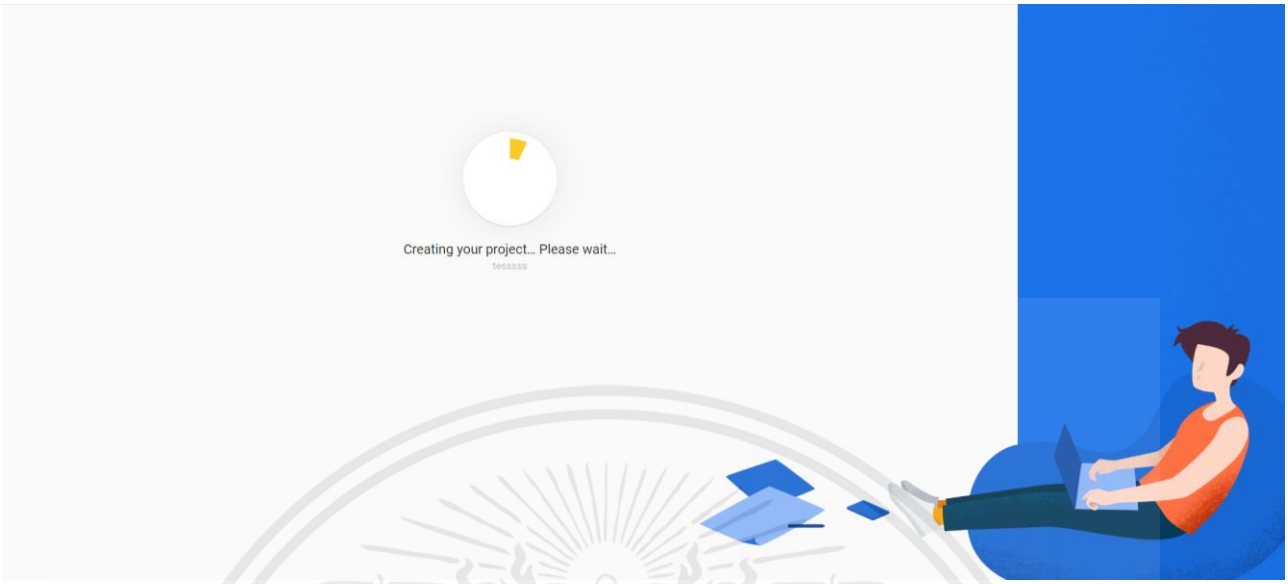
7. ทำการตั้งชื่อ Project ที่จะใช้งาน เมื่อตั้งเสร็จแล้วกด Continue



รูปที่ 3.61 รูปภาพแสดงการตั้งชื่อโปรเจค

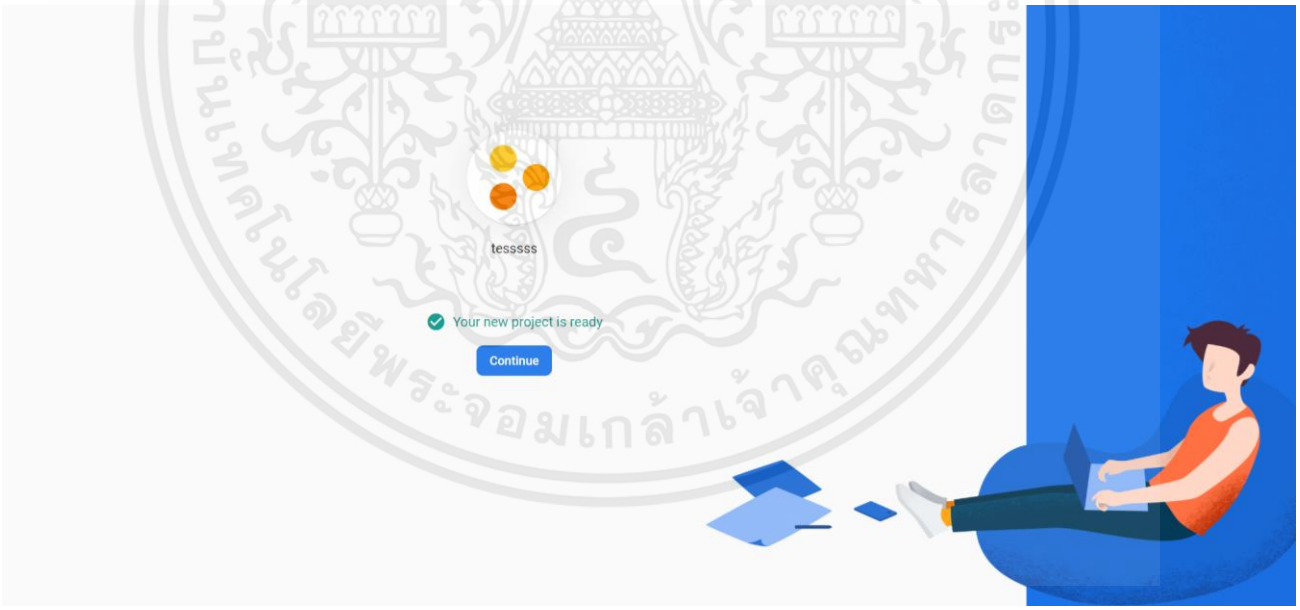
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8. ทำการกด Continue ไปเรื่อยจนขึ้นหน้าที่เพื่อรอให้ระบบทำการสร้างโปรเจคเสร็จ



รูปที่ 3.62 รูปภาพแสดงการดำเนินการสร้างโปรเจค-v’it[[

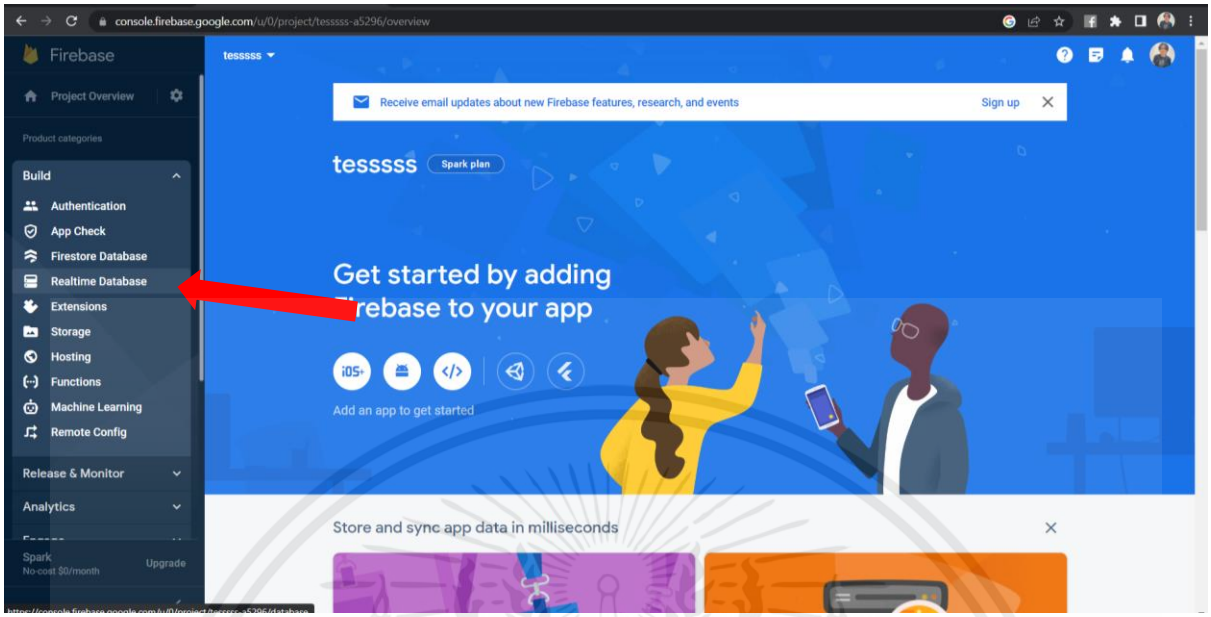
9. เมื่อรอจนระบบสร้างโปรเจคเสร็จแล้ว กด Continue



รูปที่ 3.63 รูปภาพเมื่อระบบสร้างโปรเจคสำเร็จ

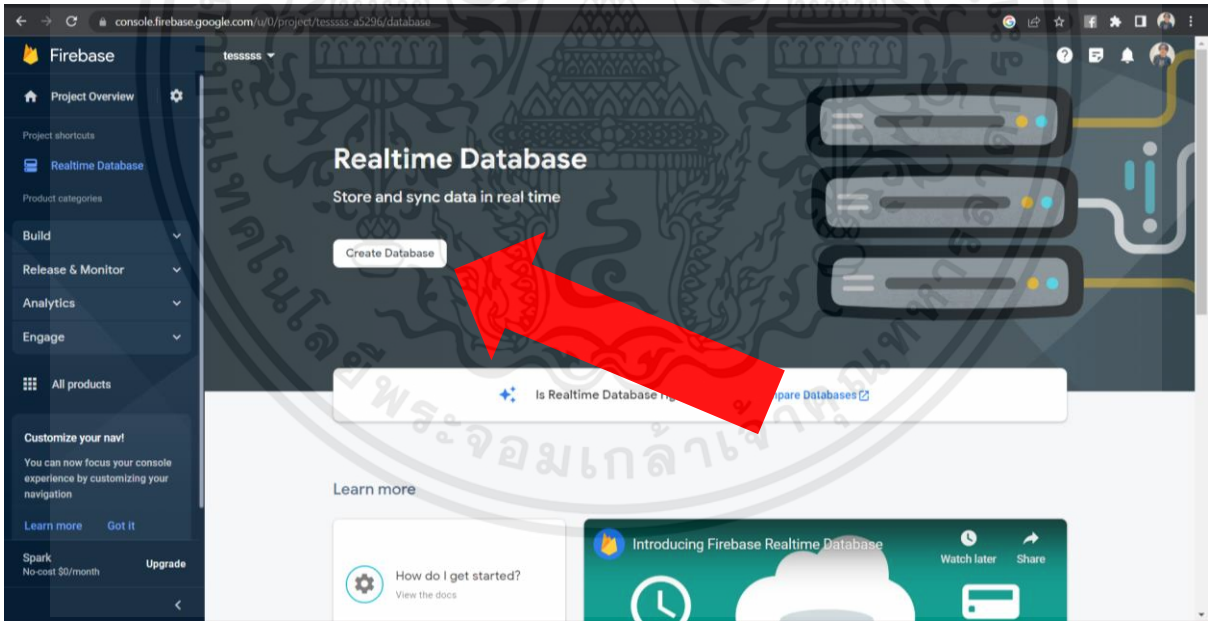
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

10. ทำการกดที่ปุ่ม Build ในแถบซ้ายมือ แล้วไปที่ Realtime database



รูปที่ 3.64 รูปภาพแสดงการเข้าใช้งานในส่วน Realtime Database

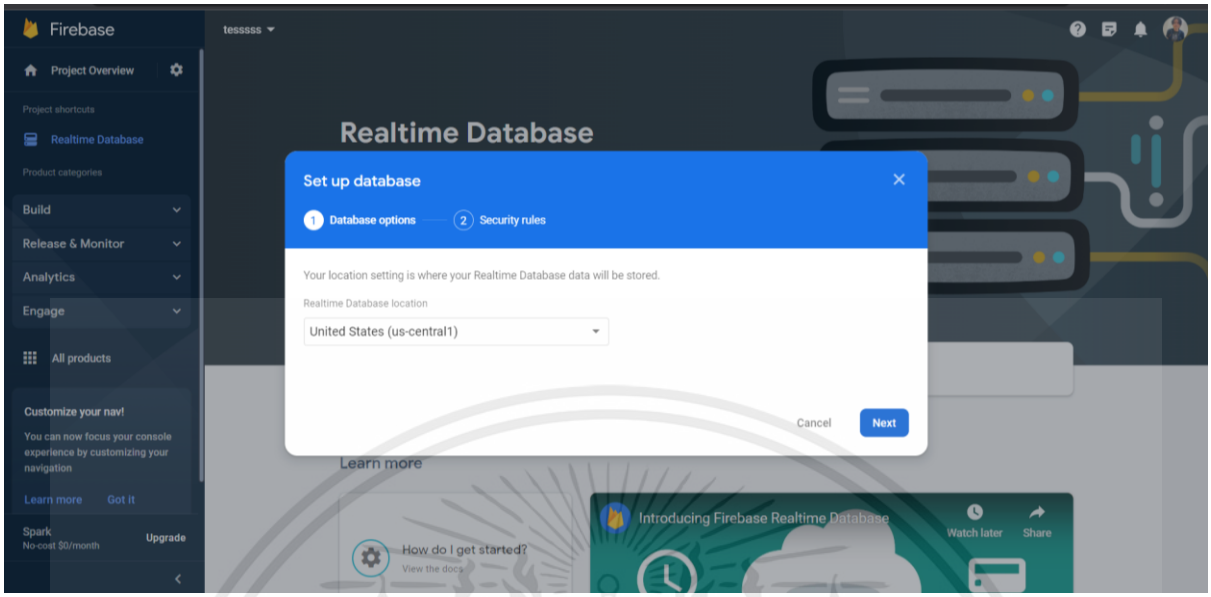
11. ในหน้า **Realtime database** กดที่ **Create database**



รูปที่ 3.65 รูปภาพแสดงตำแหน่งของปุ่ม Create database

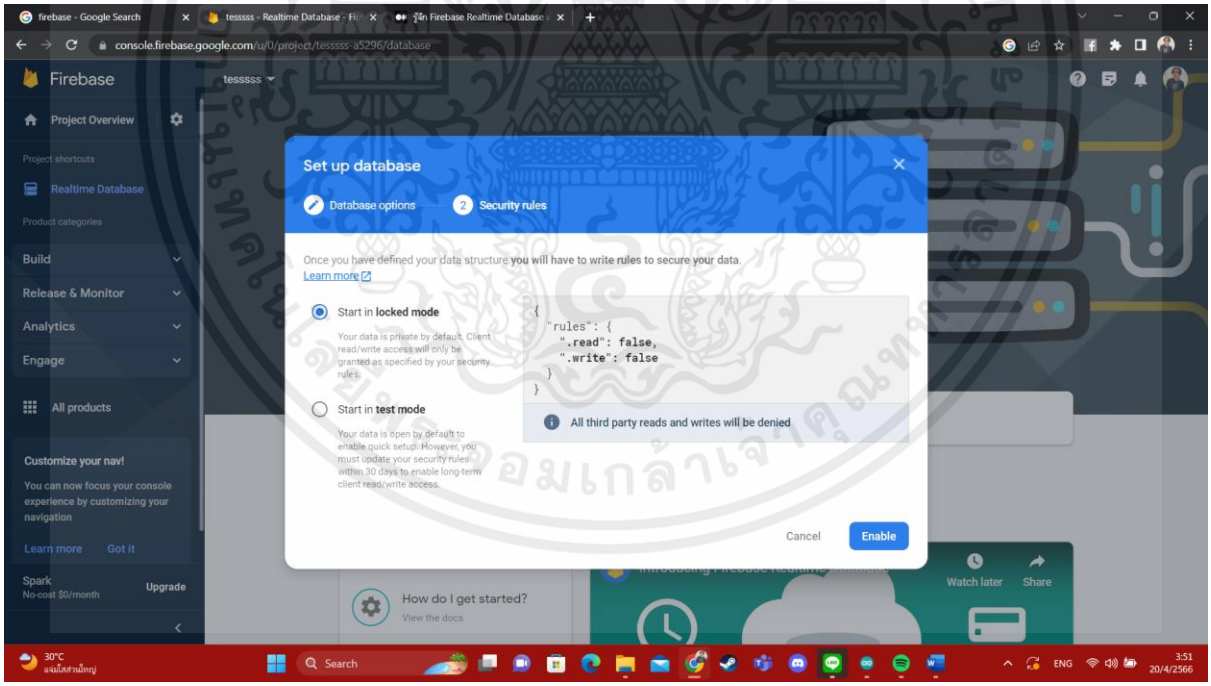
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

12. จากนั้นในหน้า Set up database ทำการกด Next



รูปที่ 3.66 รูปภาพแสดงขั้นตอนการ Set up database

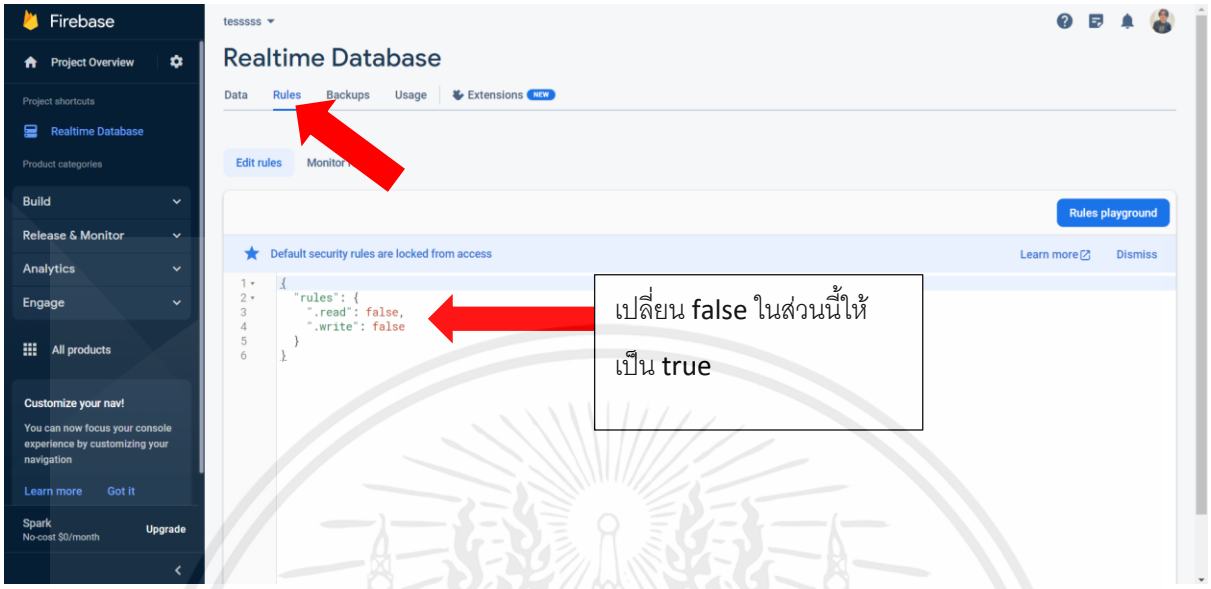
ในหน้านี้เลือกในส่วนของ Locked mode



รูปที่ 3.67 รูปภาพแสดงขั้นตอนการ Set up database ต่อ

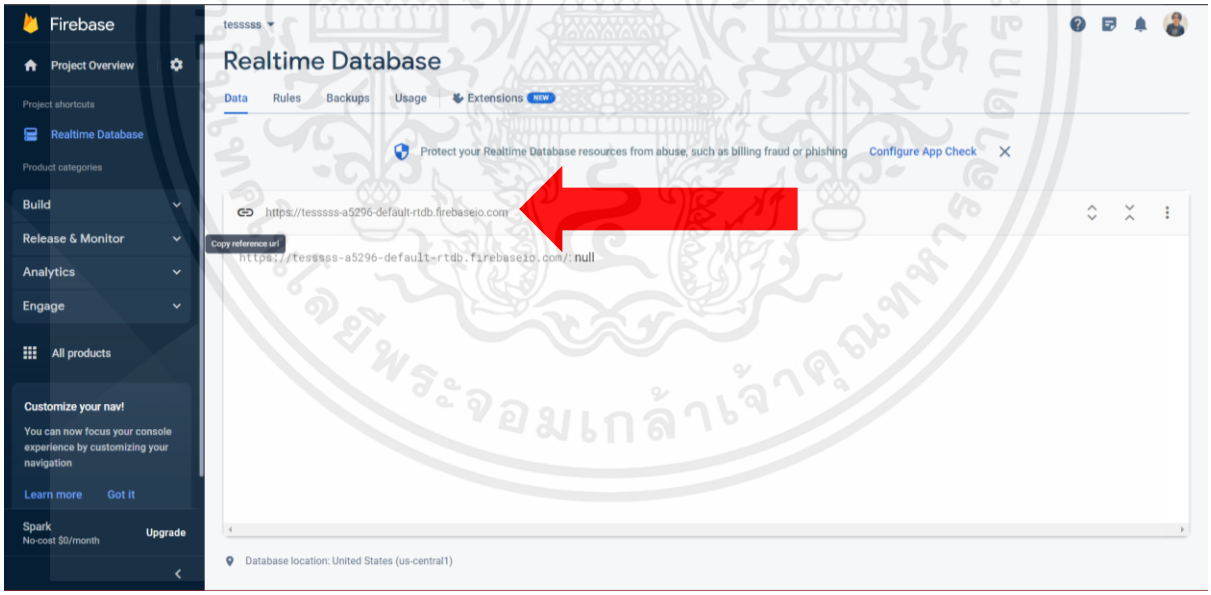
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

13. จากนั้นในแถบ Rule ให้ทำการเปลี่ยน False ใน Code ให้เป็น True เพื่ออนุญาตให้อ่านค่าและเขียนค่าได้ เป็นอันเสร็จสิ้นการ Set up firebase



รูปที่ 3.68 รูปภาพแสดงการเปลี่ยนคำสั่ง false เป็น true

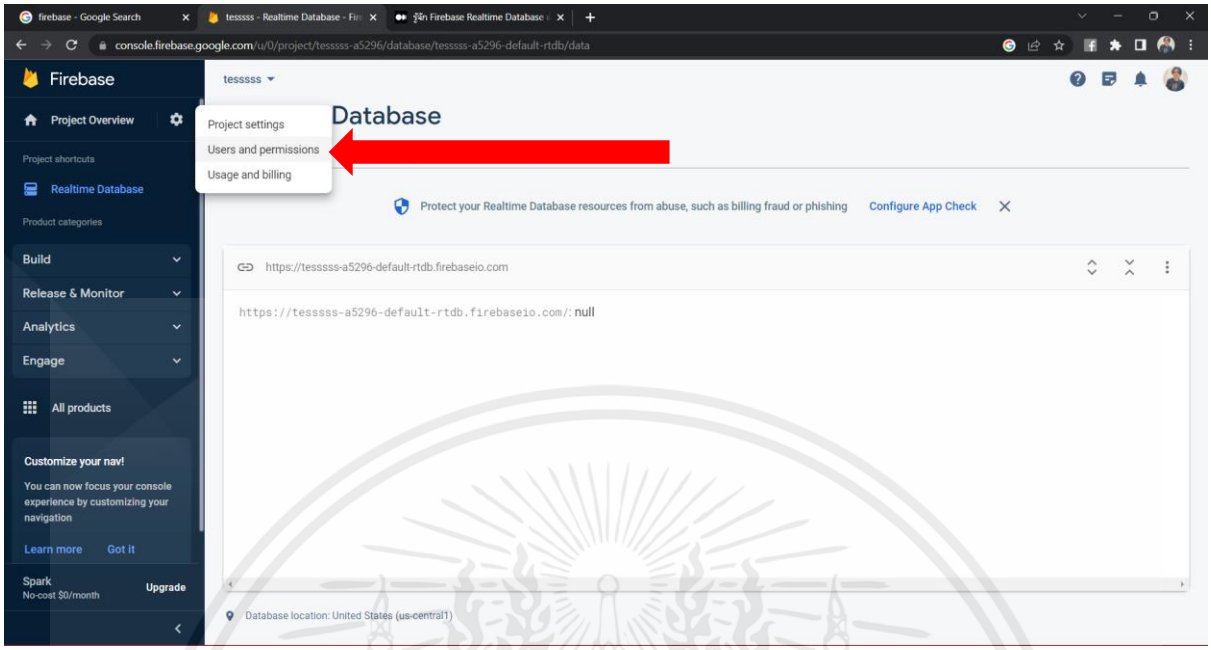
14. ในแถบ Data ทำการคัดลอก url ดังรูปที่ 3. เพื่อไปใส่ในส่วนของโปรแกรมภายหลัง



รูปที่ 3.69 รูปภาพแสดงตำแหน่ง url ของ firebase

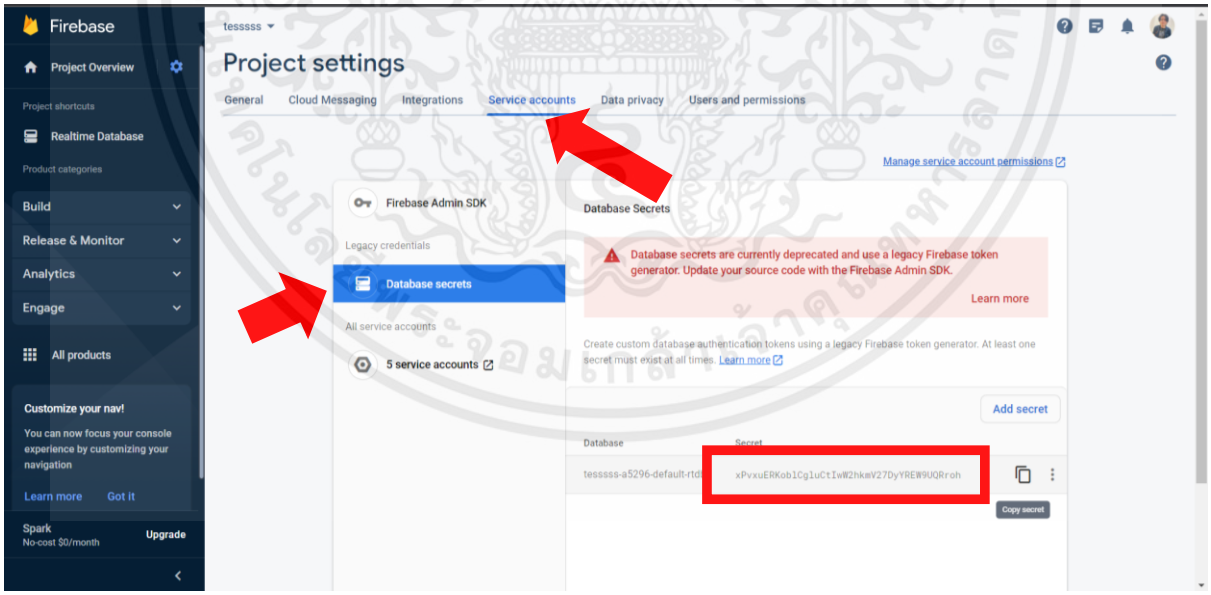
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

15. ในแถบทางซ้ายรูปตั้งค่าทำการไปที่ Users and permission



รูปที่ 3.70รูปภาพแสดงตำแหน่งของปุ่ม Users and permission

16. เมื่อเข้ามาแล้ว ไปที่แถบ Service accounts จากนั้นไปที่แถบ Database secrets แล้วทำการคัดลอก Database Secret ดังรูปที่ 3.



รูปที่ 3.71 รูปภาพแสดงตำแหน่งของ database Secret

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

17. นำ data url และ Database secret ที่ทำการคัดลอกมาในขั้นตอนที่ 14 และ 16 มาใส่ไว้ในส่วนของ API key และ Database url ในโปรแกรม Arduino IDE ดังรูปที่

```
#define API_KEY "WrBpaBAeQW6Lwgkl5DRZqASWdbVjjFOZZJBdt30w"
#define DATABASE_URL "https://project-3844738813710973354-default-rtdb.asia-southeast1.firebasedatabase.app/
```

รูปที่ 3.72 รูปภาพแสดงการใส่ API key และ Database url

18. เมื่อทำการกำหนดค่าการเชื่อมต่ออินเตอร์เน็ต และ Firebase เรียบร้อยแล้วทำการเขียนโปรแกรมส่งค่าไป Firebase และรับค่าจาก Firebase โดยมีโค้ดตัวอย่างการส่งค่าจากเซ็นเซอร์ไปที่ Firebase ดังรูปที่ 3. และการรับค่าจาก Firebase ดังรูปที่ 3.

```
////////////////////////////////////////////// ส่งค่า firebase ////////////////////////////////
  Firebase.setInt(fbdo, "data/pH", pHValue);
  Firebase.setInt(fbdo, "data/Temp", tempC);
  Firebase.setInt(fbdo, "data/TDS", tdsValue);
```

รูปที่ 3.73 รูปภาพแสดงโค้ดการส่งค่าเซ็นเซอร์ไปที่ Firebase

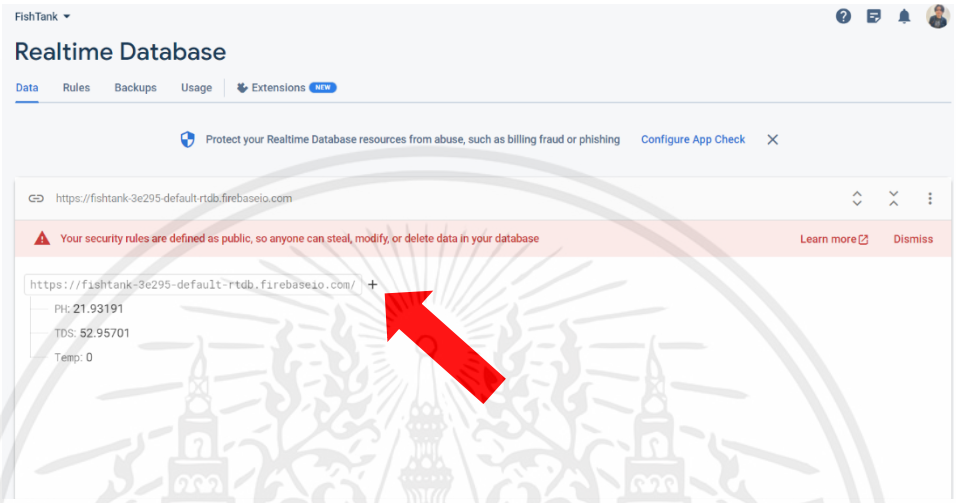
```
////////////////////////////////////////////////// fan ////////////////////////
if(Firebase.getInt(fbdo, "Relay1/fan")){
    if(fbdo.dataType() == "int") {
        Serial.println(fbdo.intData());
         fancontrol = fbdo.intData();
            if(mod == 0){
              if(fancontrol == 1){

                digitalWrite(18,HIGH);
    }
            else{
                digitalWrite(18,LOW);
    }
     }

}
}
else {
    Serial.println(fbdo.errorReason());
}
```

รูปที่ 3.74 รูปภาพแสดงโค้ดการรับค่าจาก Firebase

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

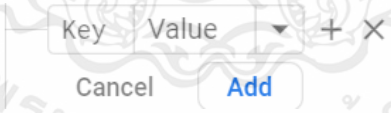
19. ในการจะรับค่าหรือส่งค่าไปยัง Firebase นั้นจะต้องกำหนดที่เอาไว้เก็บข้อมูล ว่าจะเก็บข้อมูลหรือดึงข้อมูลอะไรไปใช้โดยต้องทำการกำหนด Path ใน firebase ในที่นี้จะยกตัวอย่างการส่งค่าไปยัง Firebase โดยผ่าน path ที่กำหนดใน Firebase โดยจะมีวิธีการสร้าง path ดังนี้

1. ในหน้า Realtime database ทำการกดเครื่องหมาย + เพื่อเพิ่ม path ที่จะใช้ ดังรูปที่ 3.



รูปที่ 3.75 รูปภาพแสดงตำแหน่งปุ่มเพิ่ม path

2. เมื่อกดเพิ่มแล้วจะมีหน้าต่างมาเพื่อใส่ชื่อ Path ในช่อง Key และกำหนดชนิดตัวแปรในแถบ Value เมื่อเสร็จแล้วทำการกดปุ่ม Add



รูปที่ 3.76 รูปภาพแสดงหน้าต่างการปรับค่า Path

3. เมื่อสร้างเสร็จแล้วจะได้ดังรูปที่ 3. โดยจะสามารถเพิ่ม path ย่อยได้อีกโดยกดที่เครื่องหมาย + ข้าง Path ที่ต้องการจะสร้าง Path ย่อย และเมื่อสร้างแล้วสามารถนำ path ที่สร้างไว้ไปกำหนดในโค้ดที่ จะทำการส่งหรือรับในที่นี้จะได้ตำแหน่ง Path เป็น “data/pH” เป็นอันเสร็จสิ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

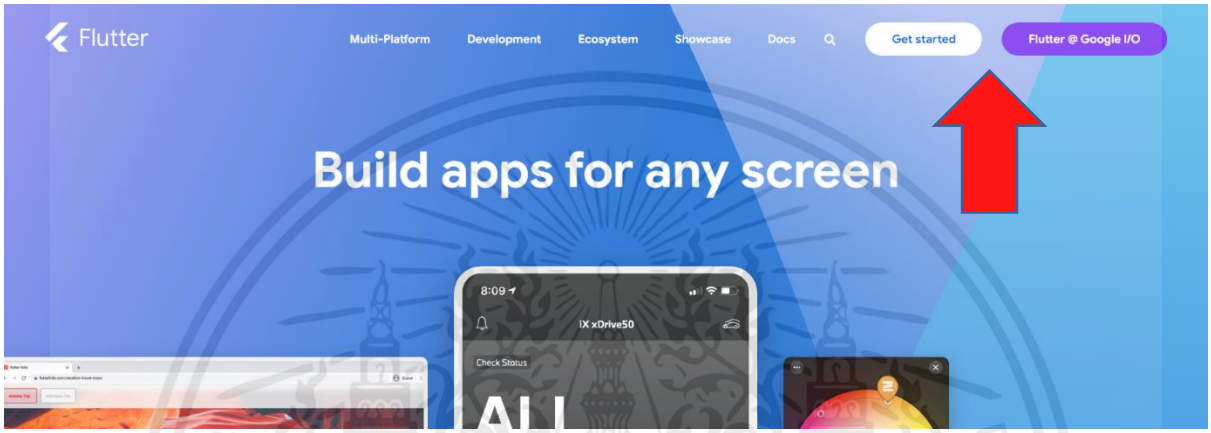
รูปที่ 3.77 รูปภาพแสดง path และ path ย่อยที่ได้ทำการสร้าง

3.12 การติดตั้งซอฟต์แวร์สำหรับการพัฒนาแอพพลิเคชั่น

ใช้ซอฟต์แวร์ Flutter SDK ในการพัฒนาแอพ ซึ่งมีวิธีการติดตั้ง ดังนี้

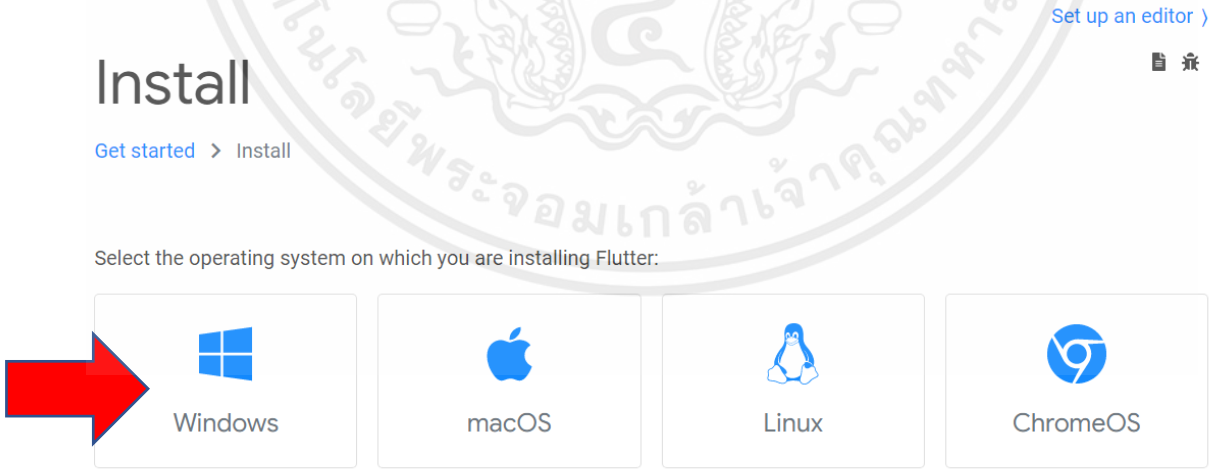
3.12.1 ดาวน์โหลดและติดตั้ง Flutter SDK

1.ไปที่เว็บไซต์ https://flutter.dev จากนั้นกด Get started



รูปที่ 3.78 หน้าเว็บไซต์ของ Flutter

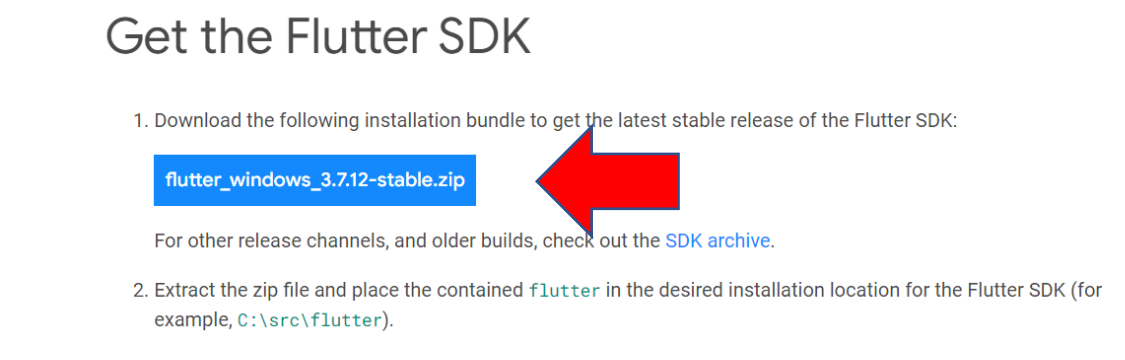
2.เลือกระบบปฏิบัติการณ์ที่ใช้ ในที่นี้จะใช้ระบบปฏิบัติการณ์ Windows



รูปที่ 3.79 ระบบปฏิบัติการณ์ของ Flutter

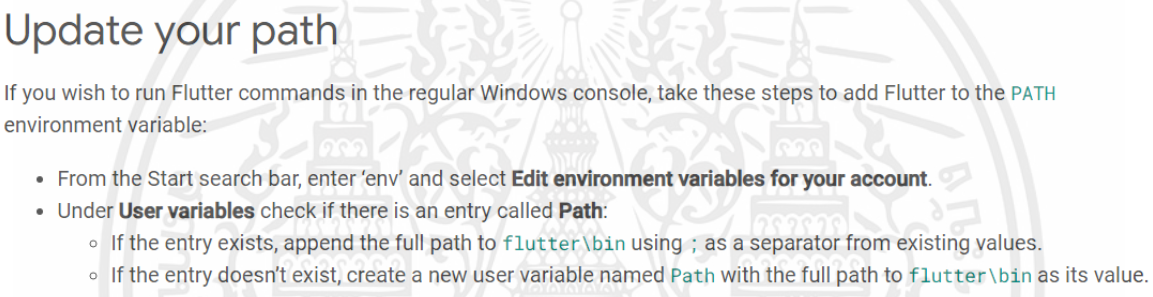
3.คลิกดาวน์โหลด Flutter และทำการแตกไฟล์เพื่อทำการติดตั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



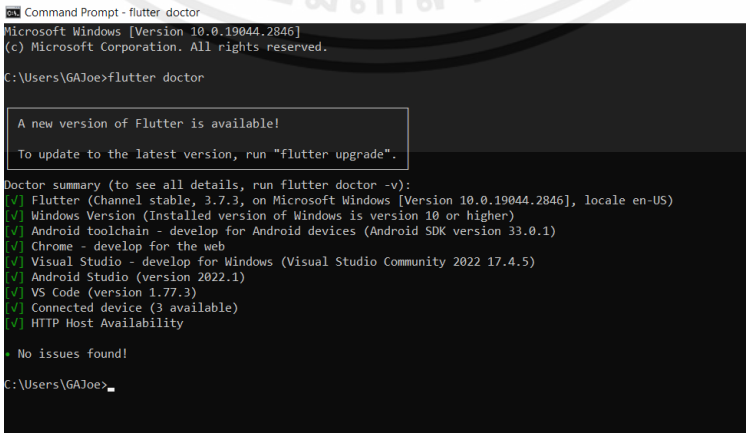
รูปที่ 3.80 ดาวน์โหลดและติดตั้ง Flutter

4.ตั้งค่า PATH environment variable ซึ่งสามารถทำตามคำแนะนำในเว็บไซต์ได้เลย



รูปที่ 3.81 ตั้งค่า PATH environment variable

5.สามารถเช็คได้ว่ามีซอฟต์แวร์ตัวไหนที่จำเป็นสำหรับการทำงานของ Flutter SDK และยังไม่ได้ติดตั้งด้วยการพิมใน cmd ว่า “flutter doctor”

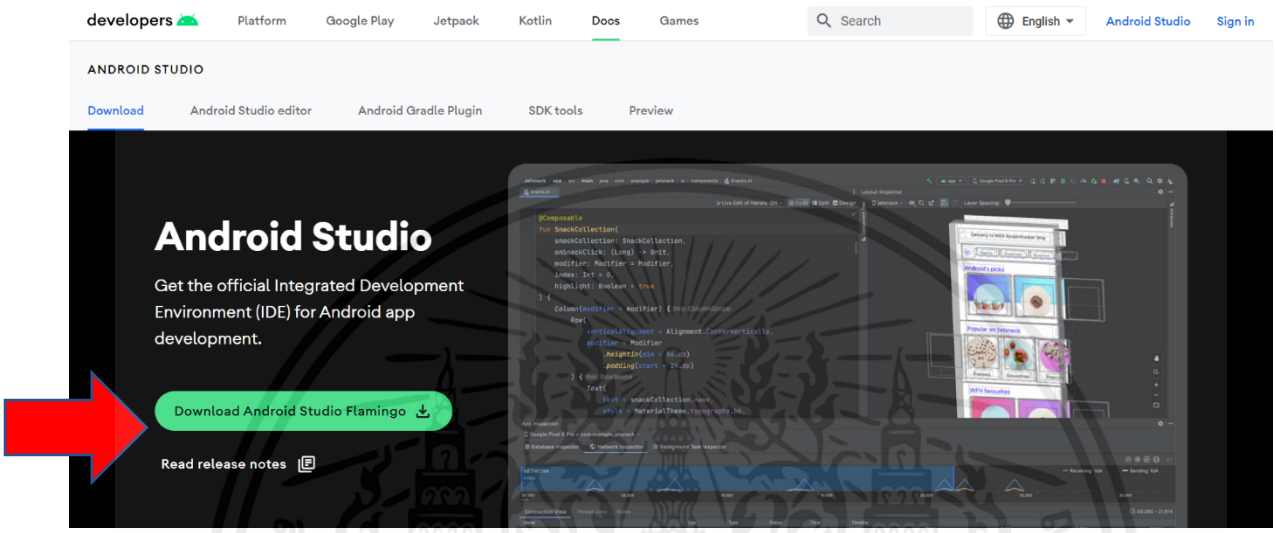


รูปที่ 3.82 เช็คซอฟต์แวร์โดยใช้คำสั่ง flutter doctor

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

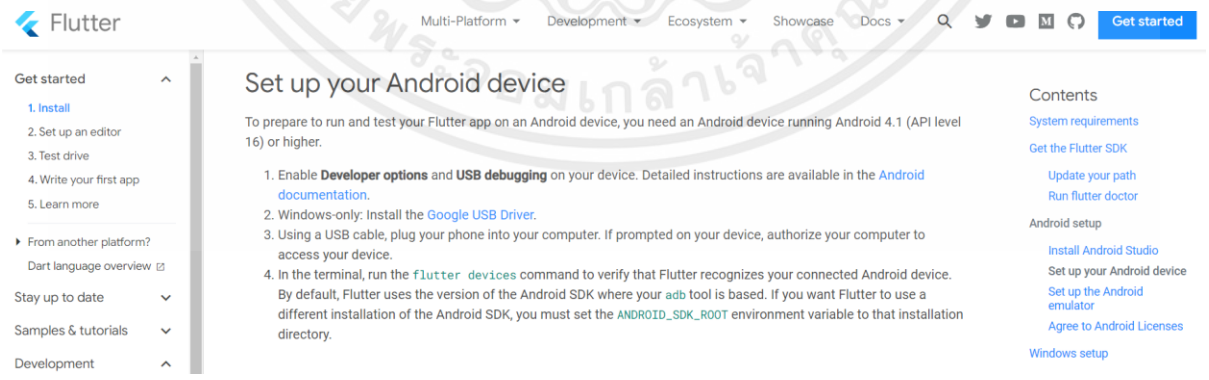
3.12.2 ดาวน์โหลดและติดตั้ง Android Studio

1.ไปที่เว็บไซต์ https://developer.android.com/studio จากนั้นกด Download Android Studio แล้วทำการติดตั้ง



รูปที่ 3.83 หน้าเว็บไซต์ของ Android Studio

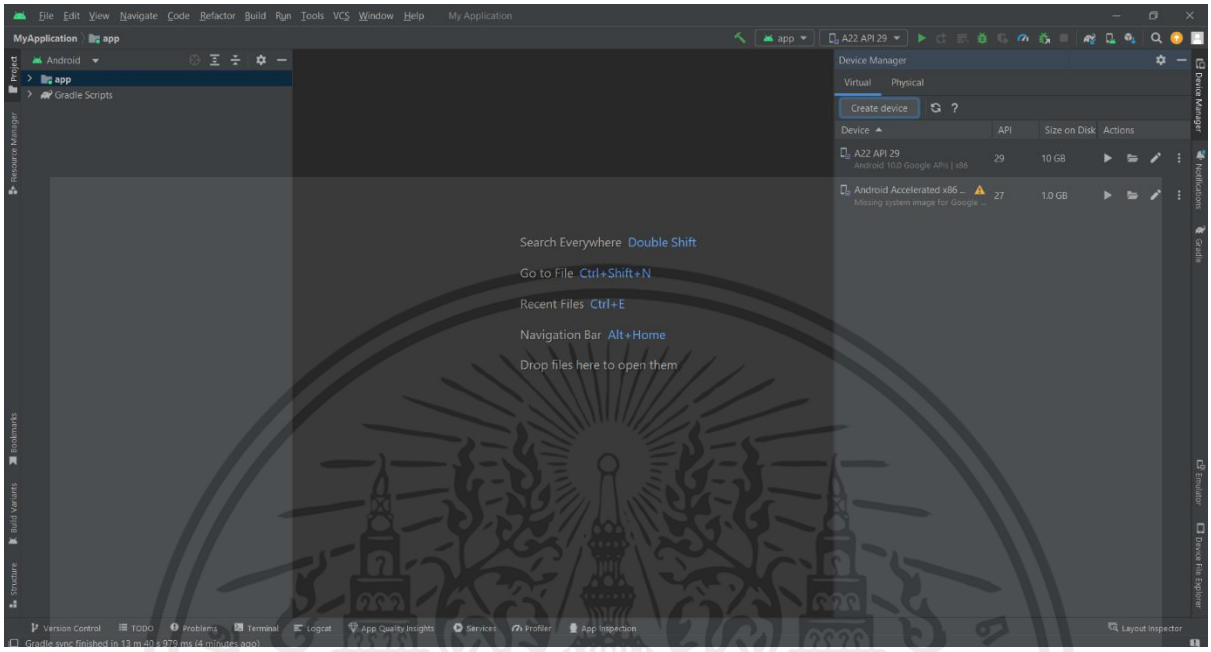
2.ทำการ Set PATH ของ Android Studio ซึ่งสามารถทำตามคำแนะนำในเว็บไซต์ของ Flutter ได้เลย



รูปที่ 3.84 ตั้งค่า PATH ของ Android Studio

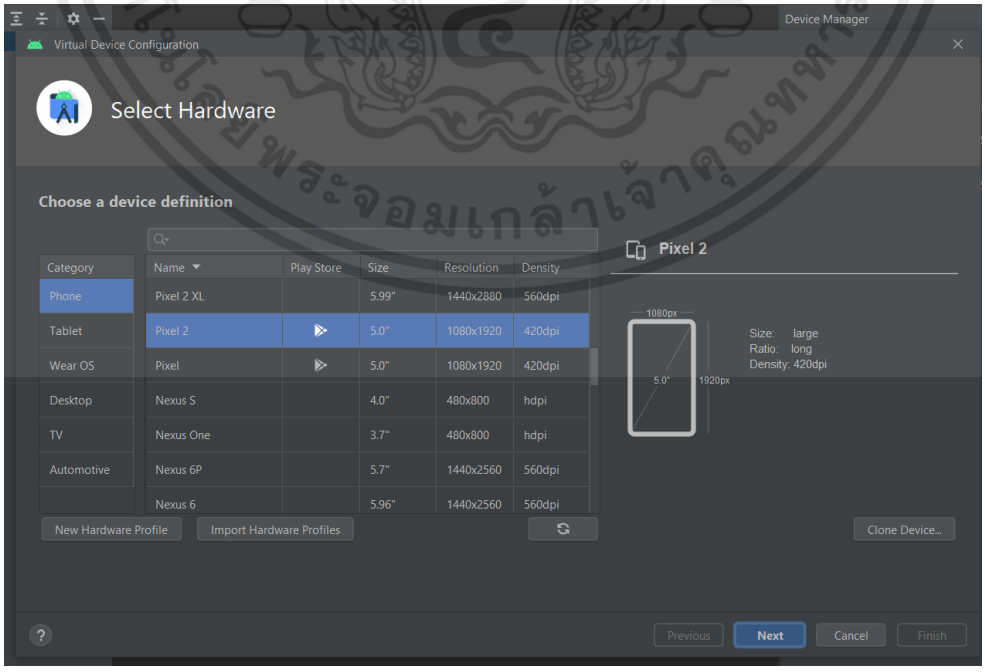
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.สร้าง Devices ที่จะเป็นตัวจำลองการทำงานของแอพพลิเคชั่น โดยสร้างใน Android Studio คลิกเข้าโปรแกรม Android Studio จากนั้นเลือก New project และเลือก create device เพื่อทำการสร้าง Devices แสดงอุปกรณ์จำลอง



รูปที่ 3.85 สร้าง Device จำลองการทำงานของแอพพลิเคชั่น

4.เลือก Hardware ปรับขนาดหน้าจอตามที่เราต้องการได้เลย

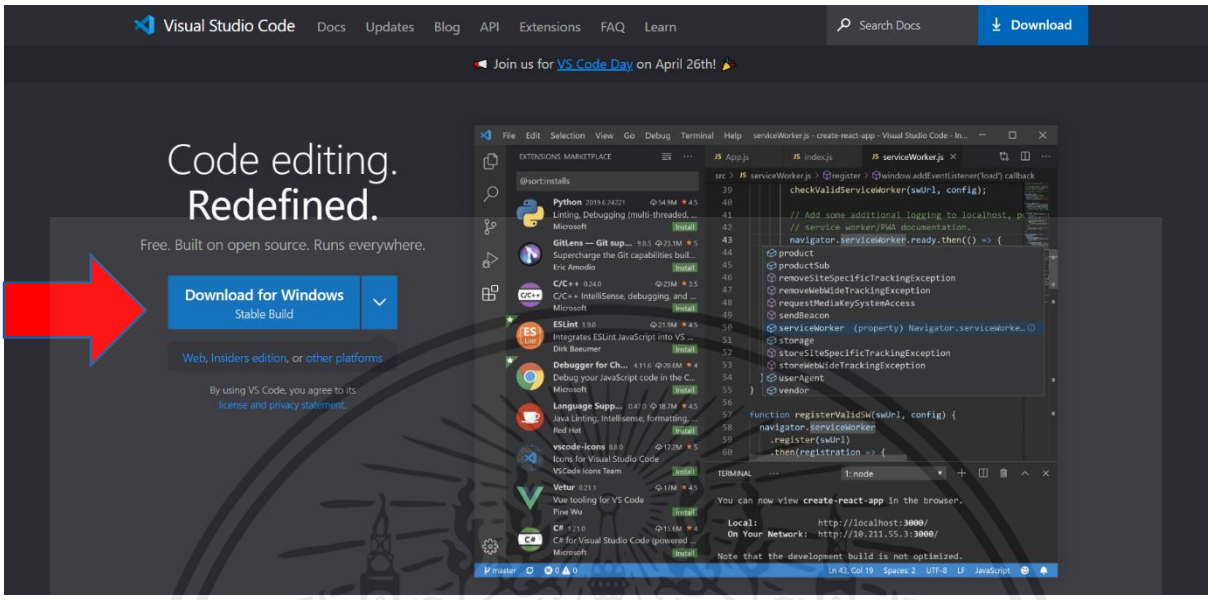


รูปที่ 3.86 ปรับขนาดหน้าจอของ Devices

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

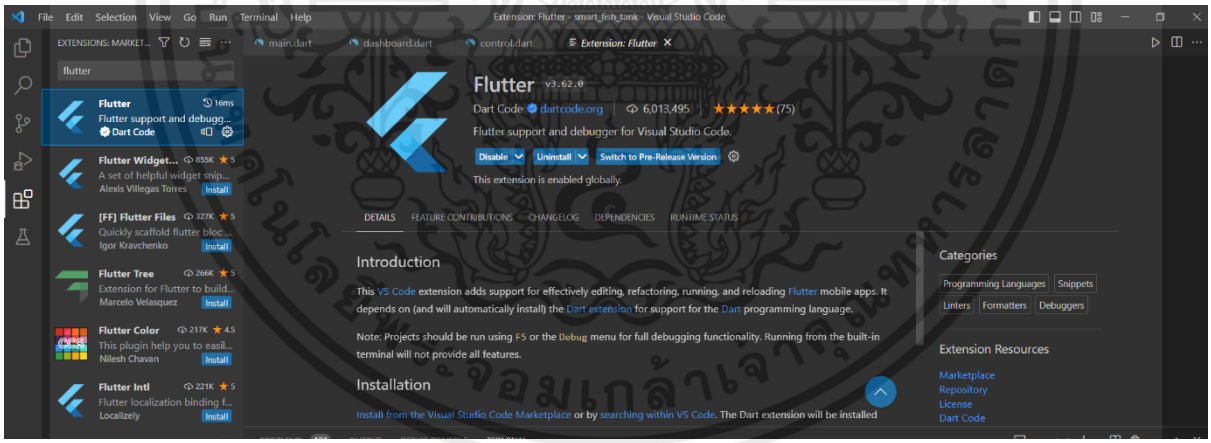
3.12.3 ดาวน์โหลดและติดตั้ง Visual Studio Code

1.ไปที่เว็บไซต์ https://code.visualstudio.com จากนั้นกด Download แล้วทำการติดตั้ง



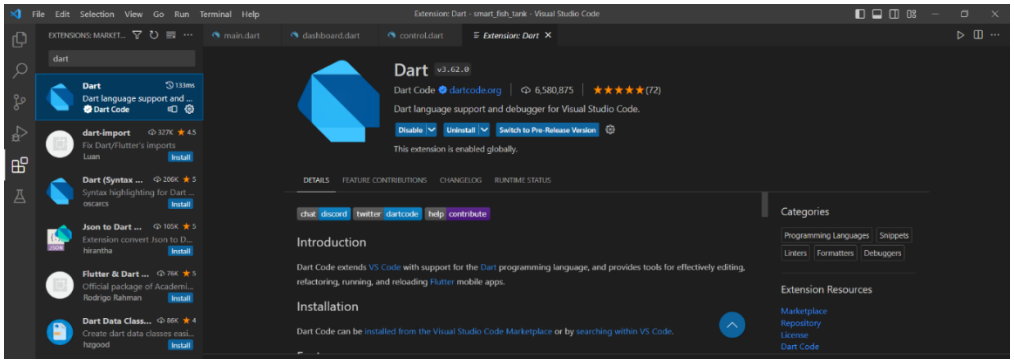
รูปที่ 3 87 หน้าเว็บไซต์ของ Visual Studio Code

2.ทำการดาวน์โหลด Flutter Extension และ Dart Extension ใน Visual Studio Code



รูปที่ 3.88 Flutter Extension ใน Visual Studio Code

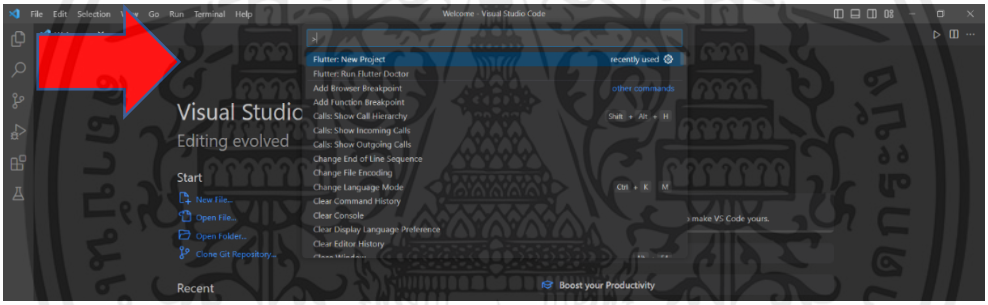
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3. 89 Dart Extension ใน Visual Studio Code

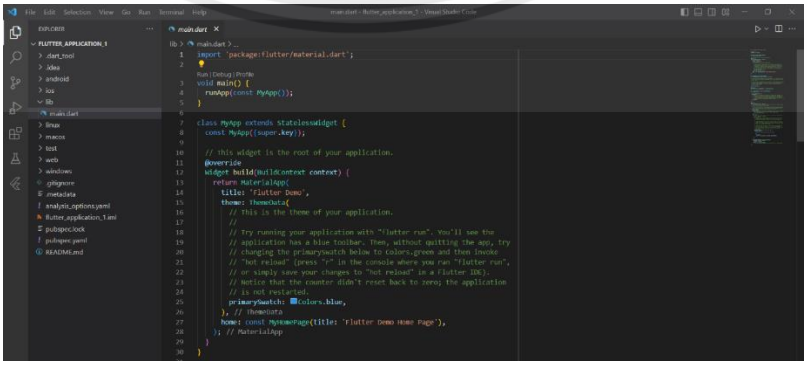
3.13 พัฒนาแอพพลิเคชั่น

1.เปิด Visual Studio Code คลิกคีย์ลัด ctrl+shift+P เพื่อทำการสร้าง Project ใหม่ขึ้นมา จากนั้นพิมพ์ว่า Flutter: New Project



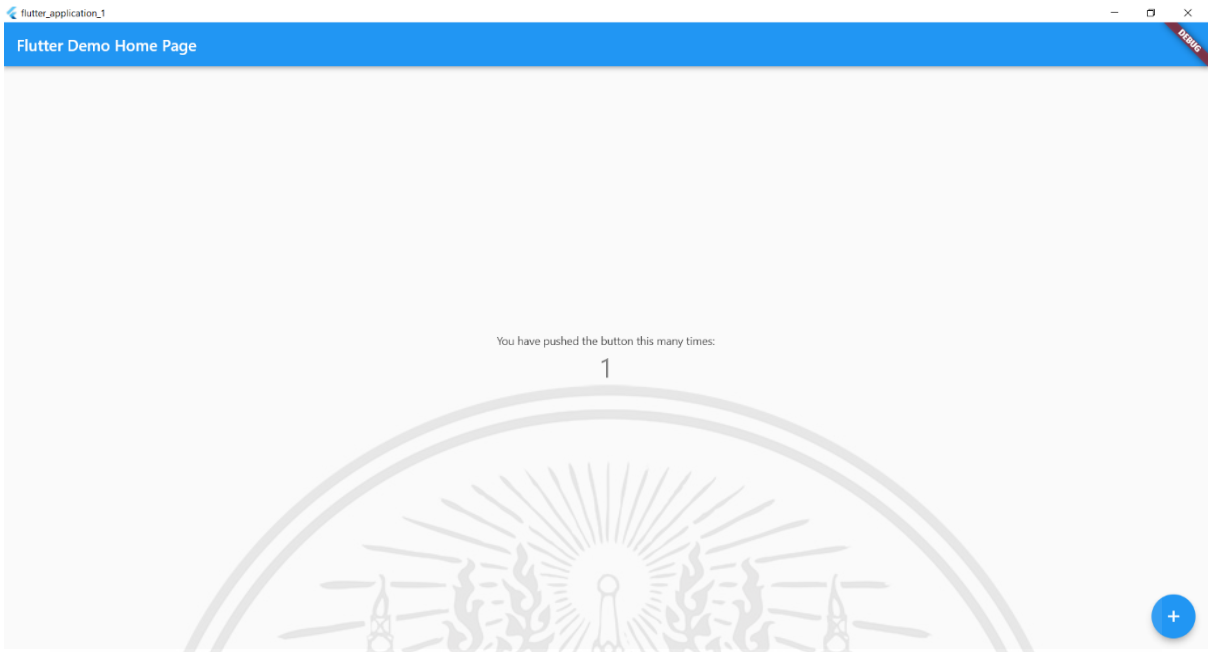
รูปที่ 3.90 สร้างโปรเจคต์ใหม่ใน Visual Studio Code

2.เลือกที่จัดเก็บโฟลเดอร์ จากนั้นตั้งชื่อโปรเจคต์ เมื่อทำเสร็จจะได้โปรเจคต์ของ Flutter ออกมา ซึ่งเป็นแอพพลิเคชั่นดั้งเดิมของ Flutter ดังรูป



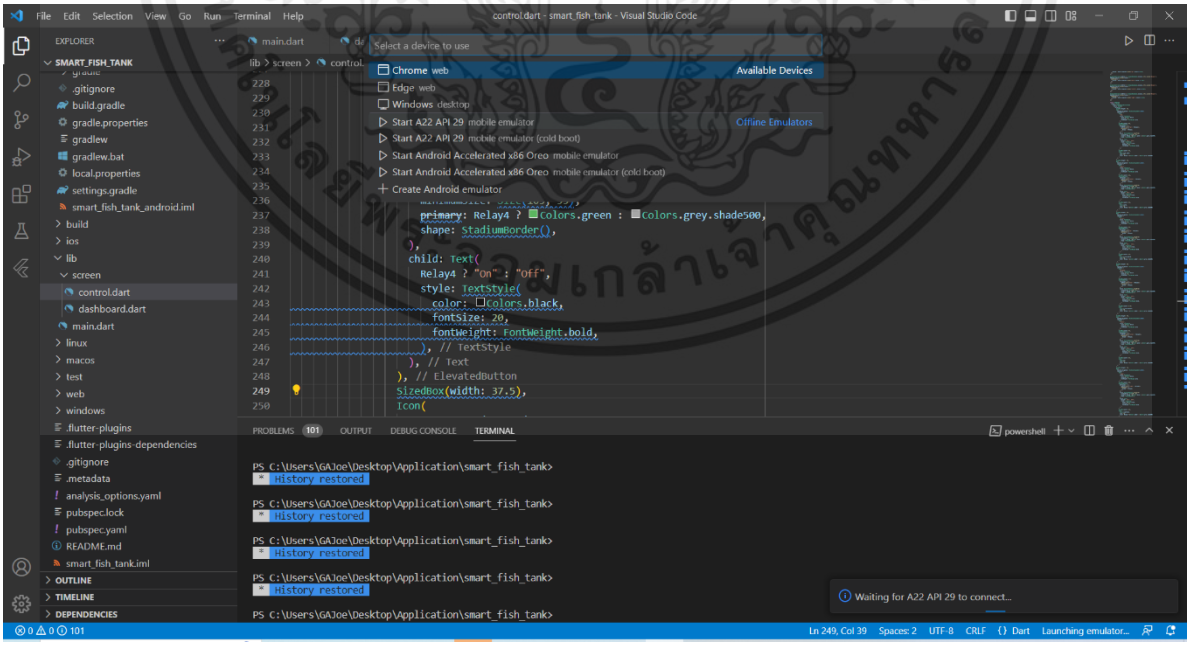
รูปที่ 3.91 โปรเจคต์ดั้งเดิมใน Flutter

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.92 แอพดั้งเดิมใน Flutter เป็นแอพที่กดปุ่มบวกแล้วเลขตรงกลางจะเพิ่ม

3.ทำการดัดแปลงแอพพลิเคชั่นจากแอพดั้งเดิมให้เป็นแอพที่เราต้องการได้เลย เมื่อเขียนโค้ดแอพพลิเคชั่นเสร็จให้เลือก Device ที่เราต้องการให้แอพของเราไปแสดง



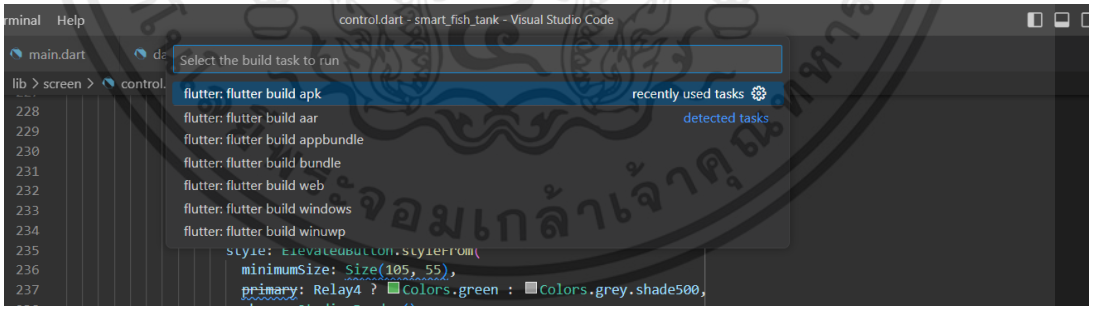
รูปที่ 3.93 เลือก Device ที่เราต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.จากนั้นกดคีย์ลัด ctrl+f5 เพื่อทำการรันแอพพลิเคชั่น

รูปที่ 3.94 ตัวอย่างแอพพลิเคชั่น

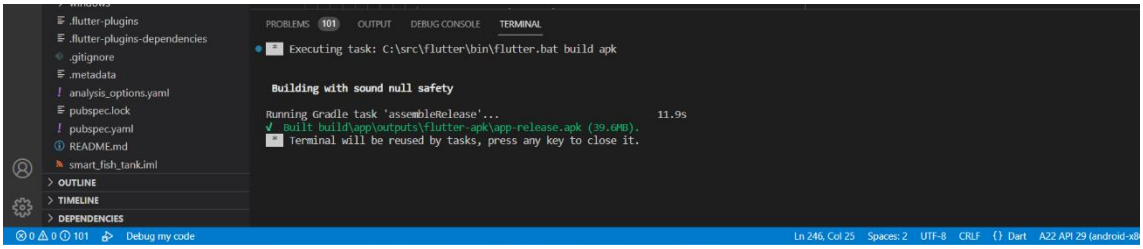
5.เมื่อพัฒนาแอพพลิเคชั่นจนเสร็จสมบูรณ์แล้ว ให้ทำการดาวน์โหลดไฟล์ .apk ไปใช้ (สำหรับระบบปฏิบัติการณ์ Android) โดยกดคีย์ลัด ctrl+shift+B จากนั้นเลือกนามสกุลไฟล์ที่เราต้องการจะดาวน์โหลด (ในที่นี้คือ .apk)



รูปที่ 3.95 สั่งโปรแกรมเพื่อทำการดาวน์โหลดไฟล์ apk

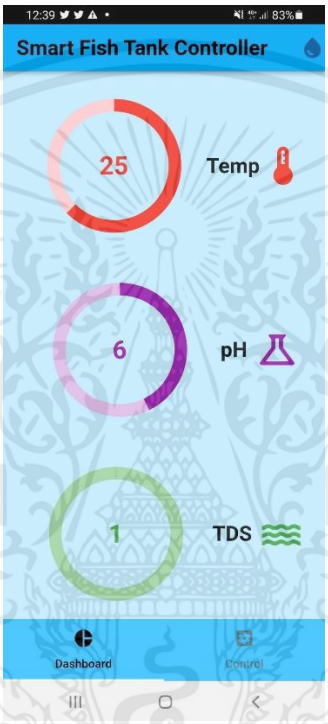
5.ที่หน้า Terminal ด้านล่างของโปรแกรม จะมีเส้นทางของไฟล์ apk ให้เรา copy Path แล้วไปหาในคอม จากนั้นเลือกดาวน์โหลดไฟล์ apk ที่ต้องการและนำไปใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.96 Path ของไฟล์ apk

6.ส่งไฟล์ apk เข้าไปในโทรศัพท์มือถือระบบปฏิบัติการ Android เป็นอันเสร็จสิ้น

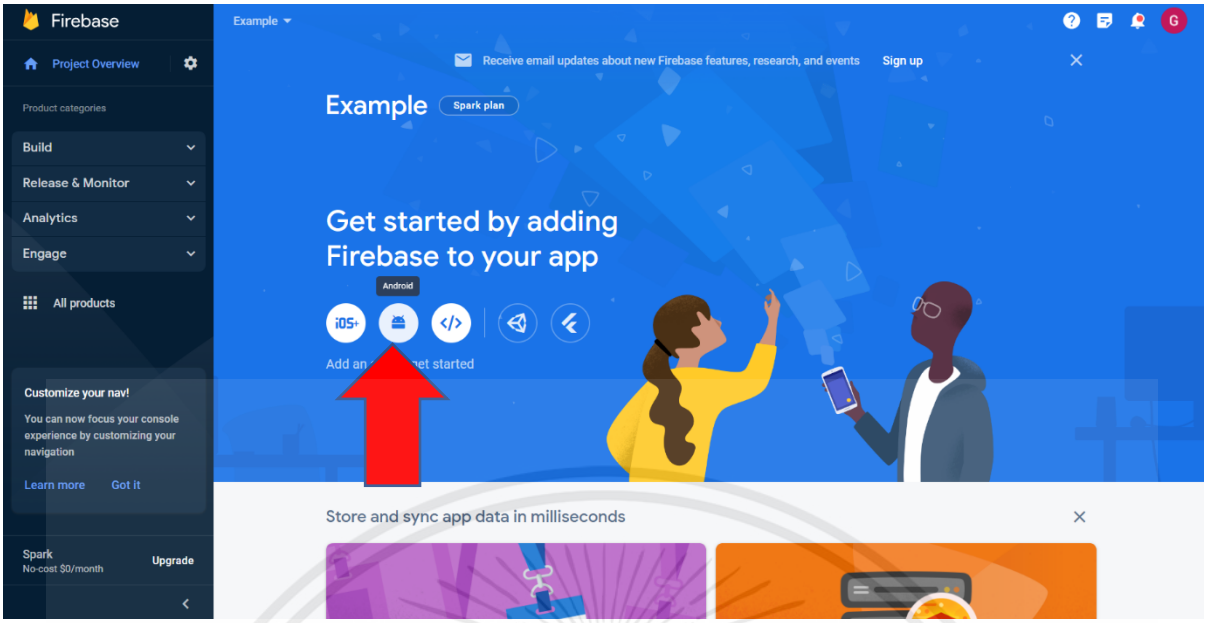


รูปที่ 3.97 แอพพลิเคชั่นในโทรศัพท์มือถือระบบปฏิบัติการณ์ Android

3.14 การส่งข้อมูลจาก Firebase มาที่แอพพลิเคชั่นของ Flutter

1.หลังจากสร้างโปรเจคต์บน Firebase ให้เลือกระบบปฏิบัติการณ์ที่เราใช้ สำหรับโปรเจคต์เลือกทำ android application จึงกดที่ android

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

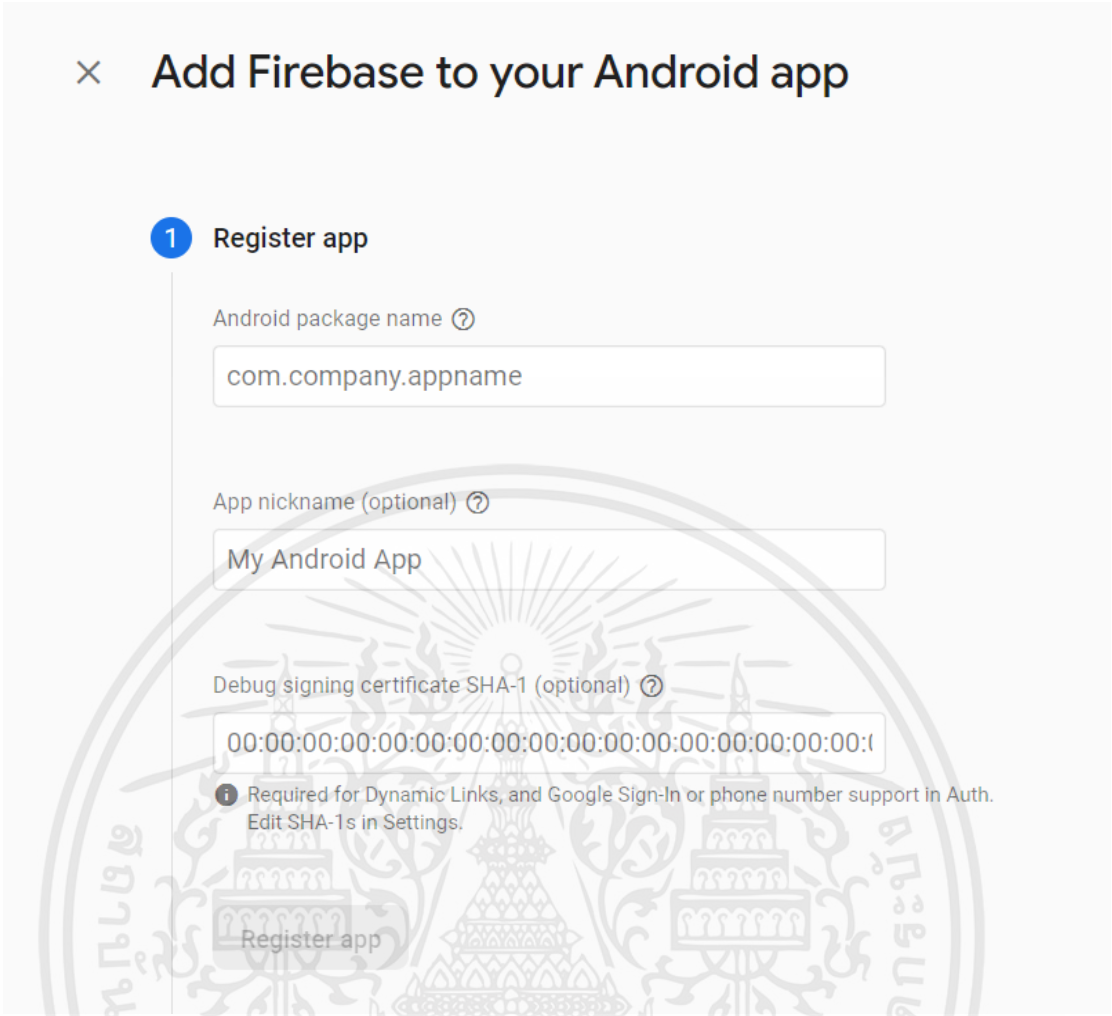


รูปที่ 3.98 เลือกระบบปฏิบัติการณ์

2.ใส่ชื่อ Android package name จากในโปรเจคต์ Flutter ของเรา และใส่ชื่อแอพลิเคชั่นของเรา ในช่อง App nickname จากนั้นกด Register app

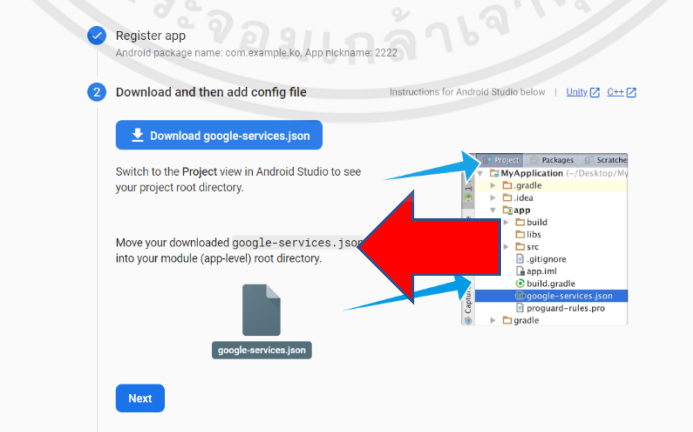
รูปที่ 3.99 ชื่อ Android package name ใน Flutter

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



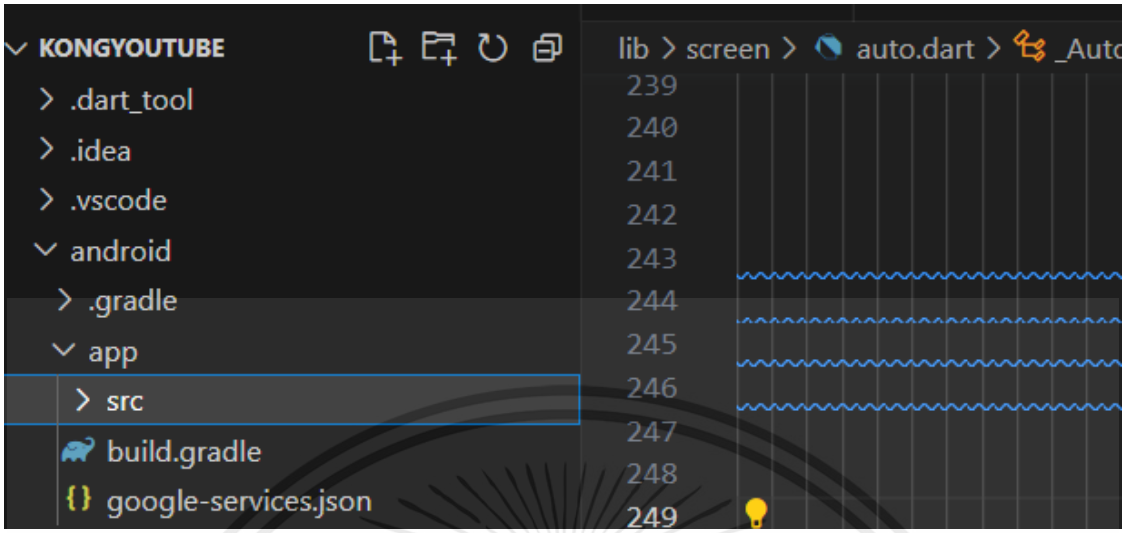
รูปที่ 3.100 ตั้งชื่อแอพพลิเคชั่นใน Firebase

3.ดาวน์โหลดไฟล์ .json แล้วนำไฟล์นี้ไปใส่ใน Flutter



รูปที่ 3.101 ดาวน์โหลดไฟล์ .json

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

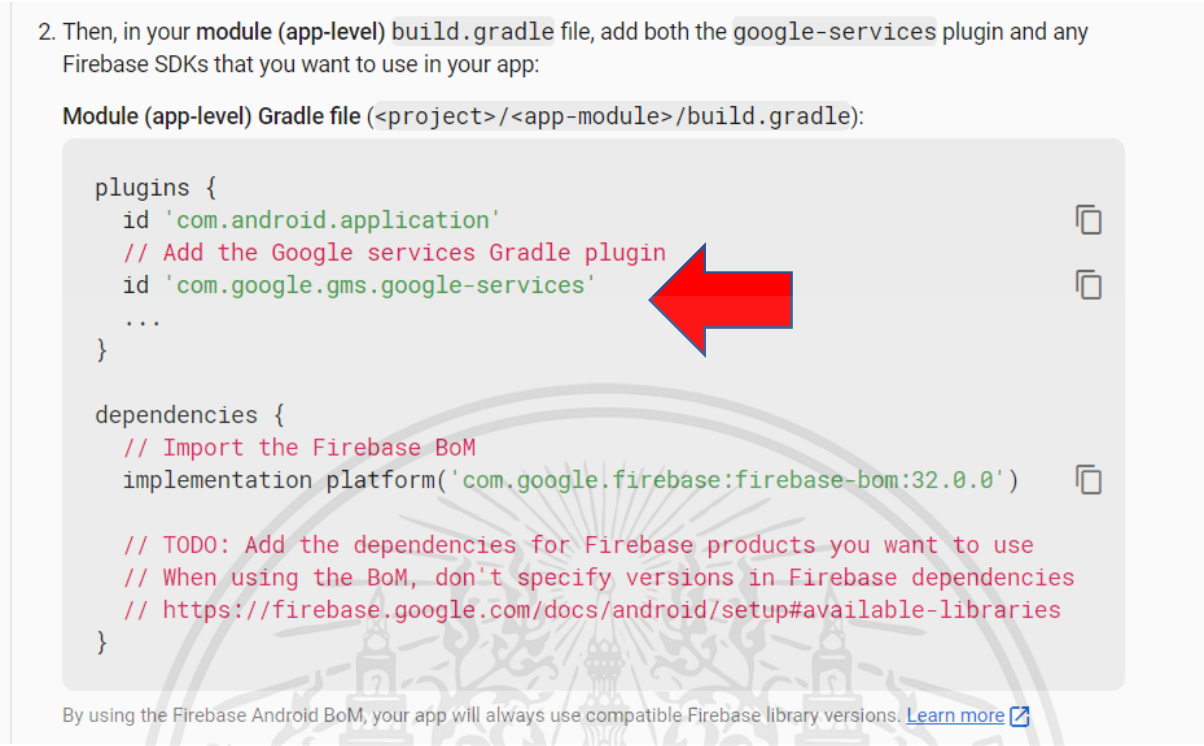


รูปที่ 3.102 นำไฟล์ไปไว้ใน app/src

4.เพิ่ม Path ตามที่ Firebase กำหนด ดังนี้

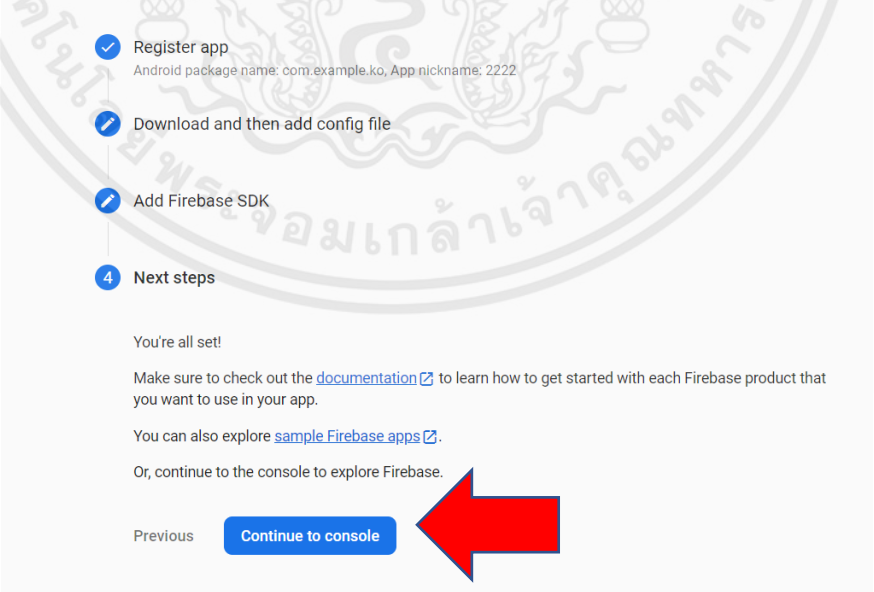
รูปที่ 3.103 เพิ่ม Path ใน android/app/build.gradle

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.104 เพิ่ม Path ใน build.gradle

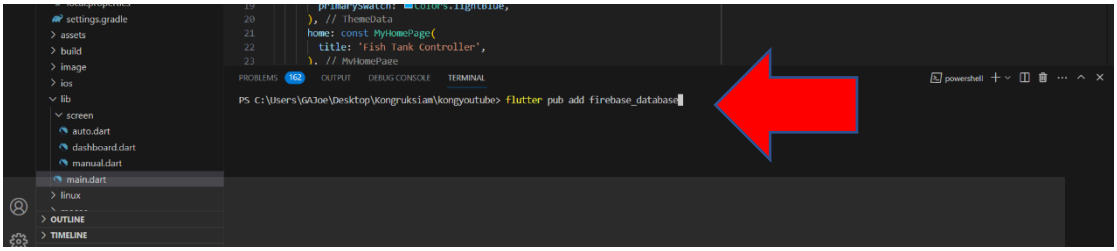
5.กด continue to console เป็นอันเสร็จสิ้น



รูปที่ 3.105 กดที่ Continue to console

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

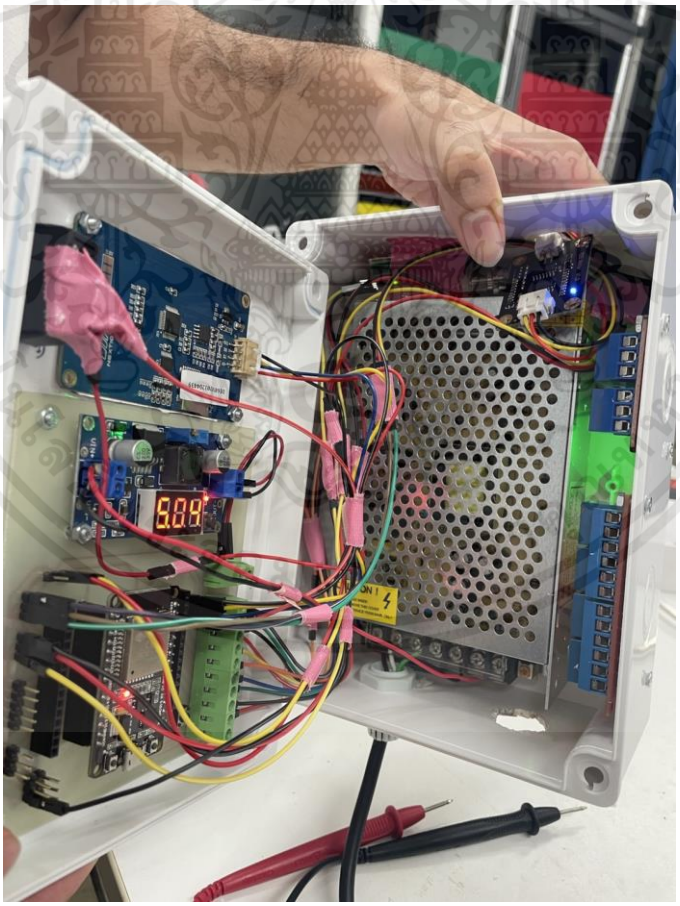
6.ใน Flutter เพิ่มตัว Extension สำหรับ Firebase และ Realtime database โดยสามารถเพิ่มได้โดยไปที่ Terminal แล้วพิมพ์ ‘flutter pub add firebase_core’ และ ‘flutter pub add firebase_database’ ตามลำดับ



รูปที่ 3.106 เพิ่มตัว Extension สำหรับ Firebase

3.15 การประกอบและรวมอุปกรณ์ต่างๆ

1. นำ Power supply, Sensor และ จอ LCD ไปประกอบในกล่องคอนโทล



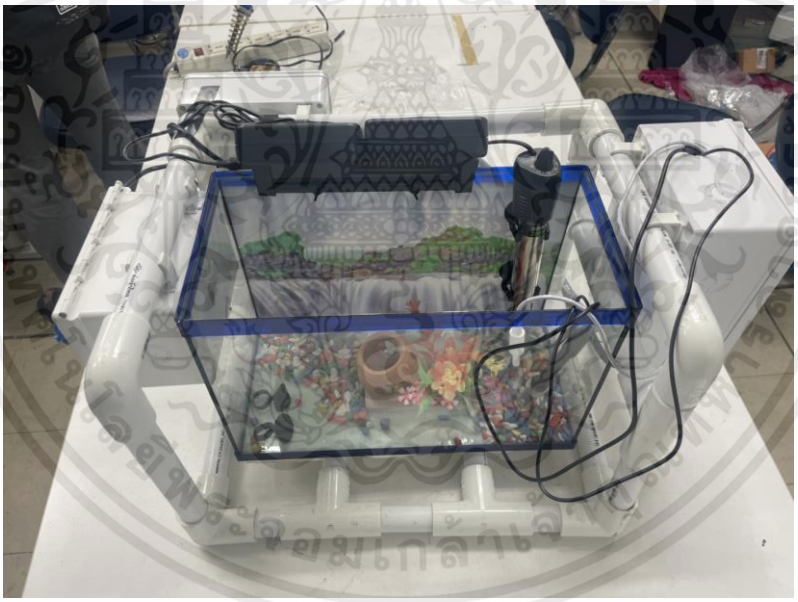
รูปที่ 3.107 รูปภาพแสดงการนำอุปกรณ์ไปประกอบในกล่องคอนโทล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. ทำการประกอบท่อ PVC เพื่อใช้เป็นโครงเอาไว้ยึดอุปกรณ์และกล่องคอนโทล และใช้วางตู้ปลาเพื่อง่ายต่อการถอดและทำความสะอาด

รูปที่ 3.108 ท่อ PVC ที่ใช้ประกอบโครงของตู้ปลา

เมื่อประกอบโครงแล้วนำกล่องคอลโทลและอุปกรณ์ต่างๆมาติดที่โครง



รูปที่ 3.109 รูปภาพเมื่อประกอบอุปกรณ์เสร็จแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

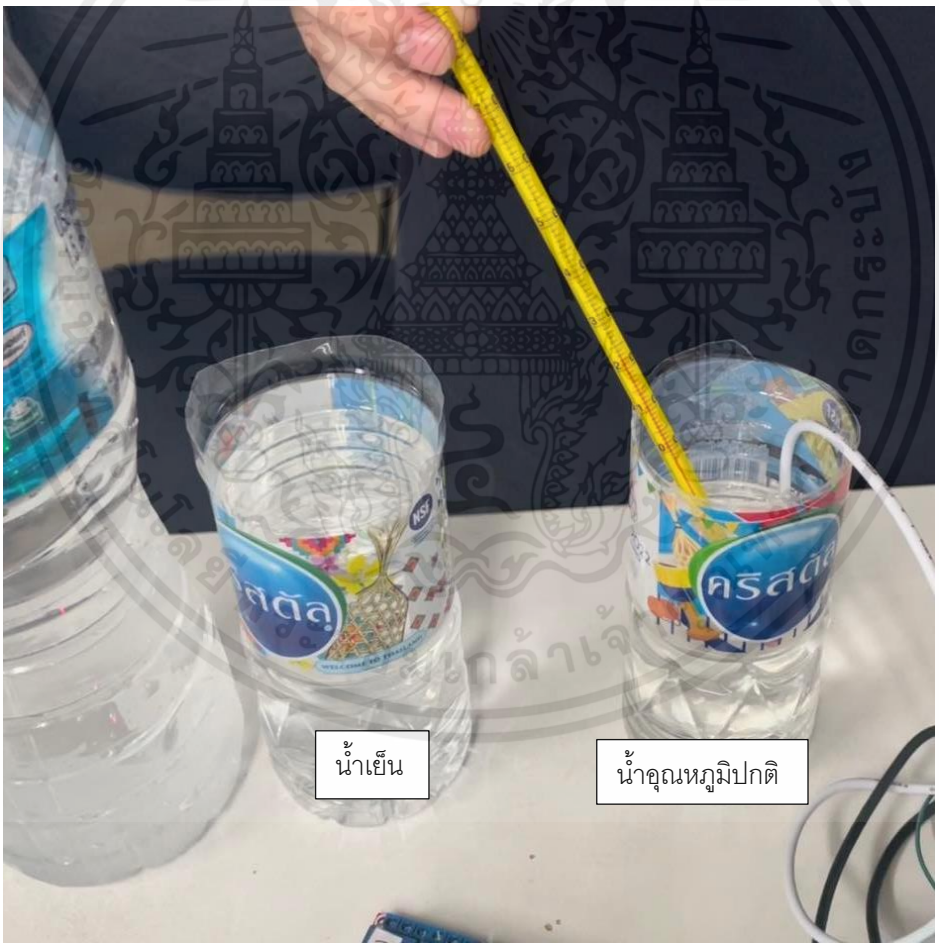
การทดลอง และผลการทดลอง

4.1 การทดลองวัดค่าจากเซ็นเซอร์มาแสดงผลบนจอภาพ

4.1.1 การทดลองวัดค่าจาก Temperature Sensor

ผู้จัดทำได้ทำการนำค่าเซ็นเซอร์ที่วัดได้มาแสดงค่าผ่านทางจอภาพ และทำการทดสอบการวัดค่าอุณหภูมิจากเซ็นเซอร์โดยมีขั้นตอนการทดลองดังนี้

1. ทำการเตรียมน้ำที่อุณหภูมิต่างกัน 2 ตัวอย่าง คือ น้ำที่มีอุณหภูมิปกติ กับน้ำเย็น

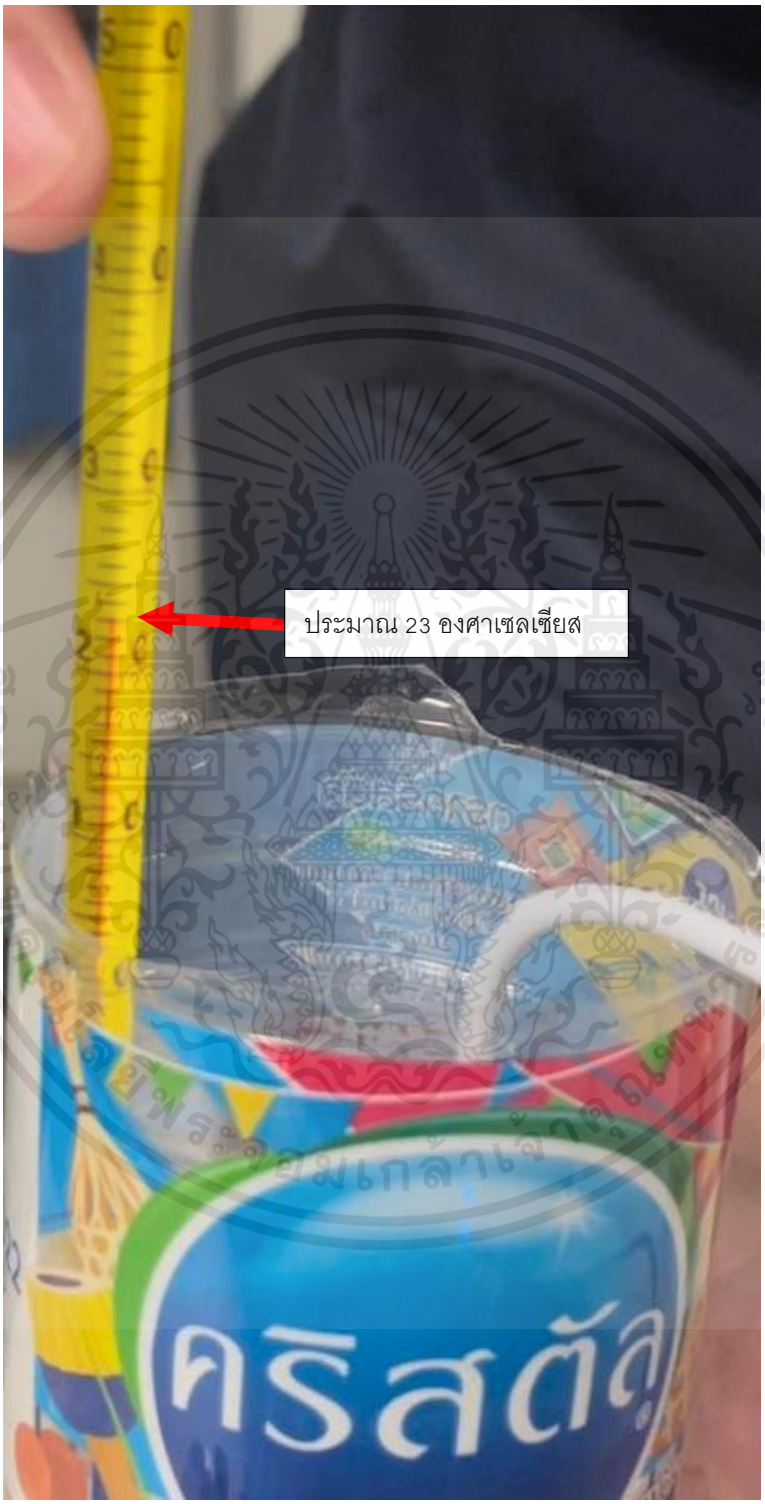


รูปที่ 4. 1 ภาพของน้ำที่ใช้ทำการทดลอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. ทำการวัดค่าเทียบกับเทอร์โมมิเตอร์ของน้ำทั้งสอง

วัดอุณหภูมิของน้ำในอุณหภูมิปกติด้วยเทอร์โมมิเตอร์ มีค่าประมาณ 23 องศาเซลเซียส



รูปที่ 4.2 ภาพขณะวัดอุณหภูมิของน้ำอุณหภูมิปกติด้วยเทอร์โมมิเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วัดอุณหภูมิของน้ำเย็นด้วยเทอร์โมมิเตอร์ มีค่าประมาณ 17 องศาเซลเซียส

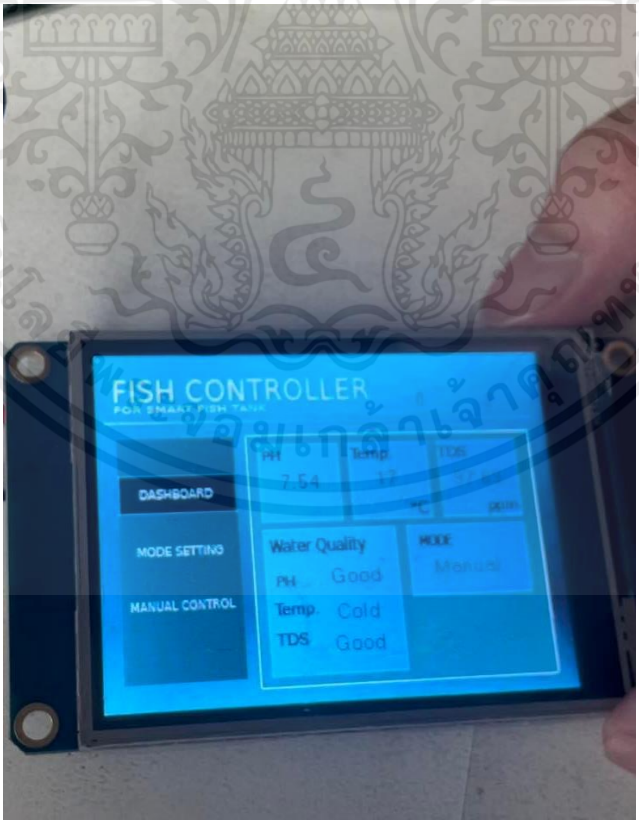
รูปที่ 4.3 ภาพขณะวัดอุณหภูมิของน้ำเย็นด้วยเทอร์โมมิเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วัดอุณหภูมิของน้ำในอุณหภูมิปกติด้วย Temperature sensor มีค่าประมาณ 23 องศาเซลเซียส ซึ่งมีค่าตรงกับอุณหภูมิที่วัดได้กับเทอร์โมมิเตอร์

รูปที่ 4.4 ภาพขณะวัดอุณหภูมิของน้ำอุณหภูมิปกติด้วยเซนเซอร์

วัดอุณหภูมิของน้ำเย็นด้วย Temperature sensor มีค่าประมาณ 17 องศาเซลเซียส ซึ่งมีค่าตรงกับอุณหภูมิที่วัดได้กับเทอร์โมมิเตอร์



รูปที่ 4.5 ภาพขณะวัดอุณหภูมิของน้ำเย็นด้วยเซนเซอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.2 การทดลองวัดค่าจาก pH Sensor

ผู้จัดทำได้ทำการนำค่าเซ็นเซอร์ที่วัดได้มาแสดงค่าผ่านทางจอภาพ และทำการทดสอบการวัดค่า pH จากเซ็นเซอร์โดยมีขั้นตอนการทดลองดังนี้

1. ทำการเตรียมน้ำขนาด 250ml มาผสมกับผง pH buffer ทั้งหมด 3 ค่าคือ 4.01, 6.86 และ 9.18

รูปที่ 4.6 ภาพของน้ำที่มี pH ค่าต่างๆ

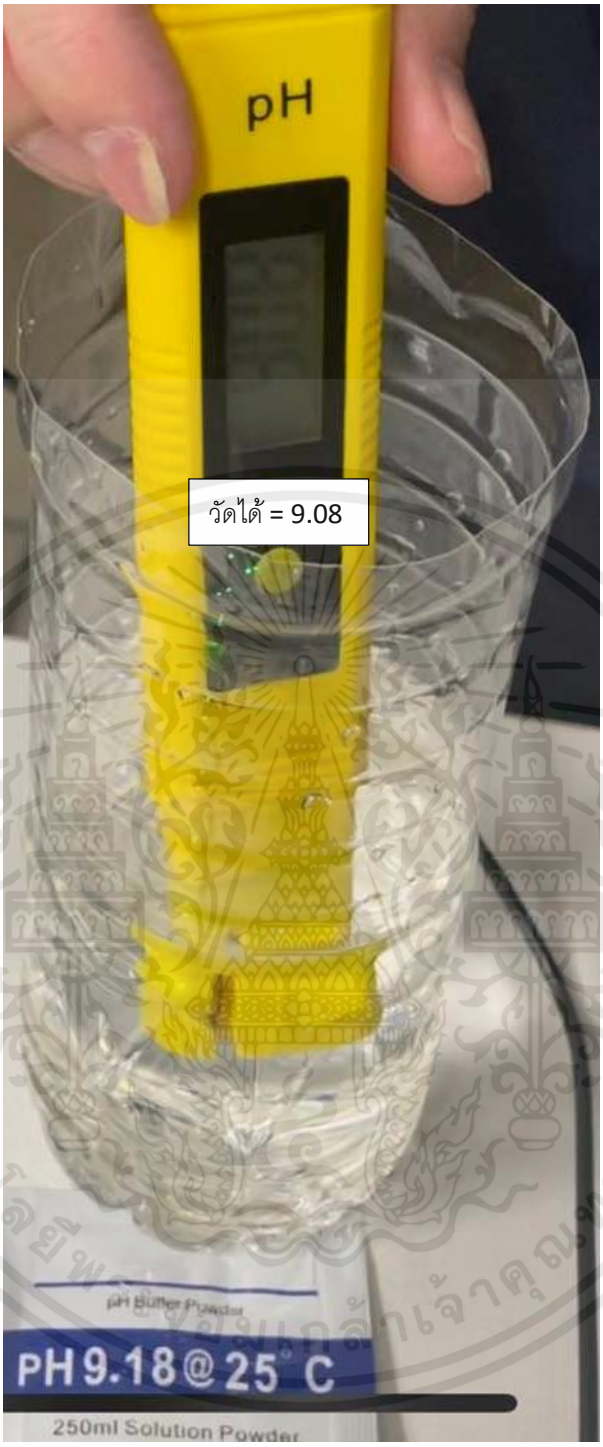
2. ทำการทดสอบวัดค่า pH โดยใช้เครื่องมือวัด pH แบบมาตรฐาน ทั้ง 3 ค่า

รูปที่ 4.7 ภาพขณะวัด pH = 6.86 ด้วยเครื่องมือวัด pH

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.8 ภาพขณะวัด pH = 4.01 ด้วยเครื่องมือวัด pH

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.9 ภาพขณะวัด pH = 9.18 ด้วยเครื่องมือวัด pH

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ทำการทดสอบวัดค่า pH โดยใช้ pH sensor ทั้ง 3 ค่า

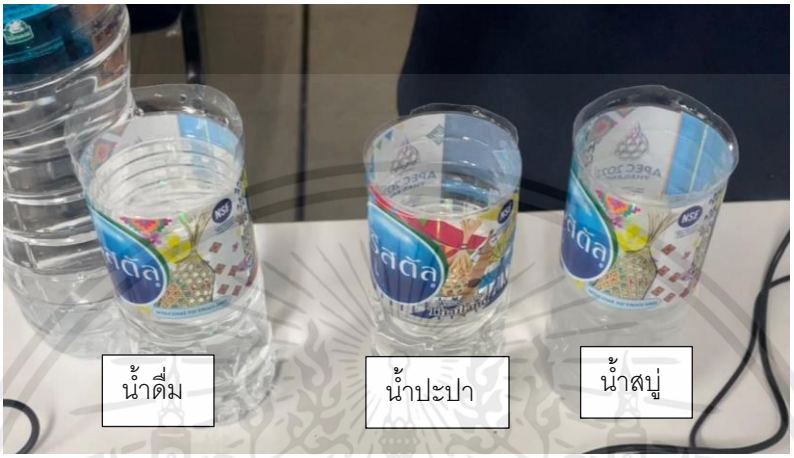
รูปที่ 4.10 ภาพขณะวัด pH ด้วย pH Sensor

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.3 การทดลองวัดค่าจาก TDS sensor

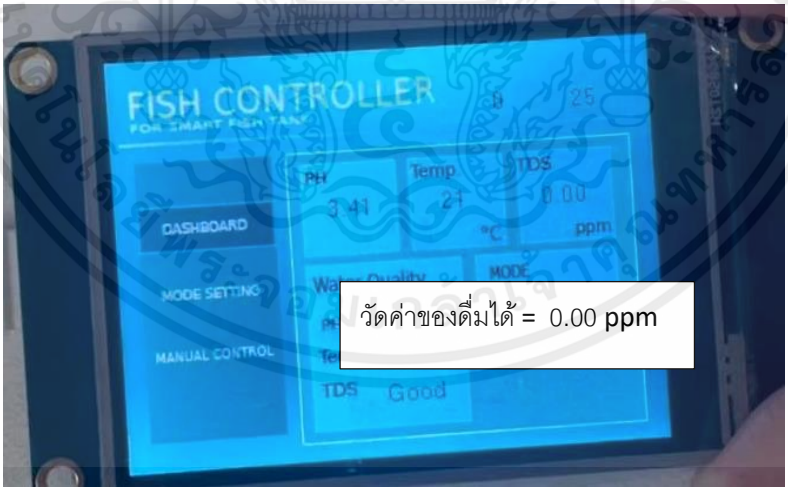
ผู้จัดทำได้ทำการนำค่าเซ็นเซอร์ที่วัดได้มาแสดงค่าผ่านทางจอภาพ และทำการทดสอบการวัดค่า TDS จากเซ็นเซอร์โดยมีขั้นตอนการทดลองดังนี้

1. ทำการเตรียมน้ำที่มีค่าความขุ่นต่างกัน 3 แบบ คือ น้ำดื่ม น้ำประปา และน้ำที่ผสมสบู่



รูปที่ 4.11 ภาพของน้ำ 3 ชนิดที่ต่างกัน

2. ทำการทดสอบวัดค่า tds โดยใช้ tds sensor ของน้ำทั้ง 3 แบบ



รูปที่ 4.12 ภาพขณะวัดค่า TDS ของน้ำดื่มด้วย TDS Sensor

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.13 ภาพขณะวัดค่า TDS ของน้ำปะปาด้วย TDS Sensor

รูปที่ 4.14 ภาพขณะวัดค่า TDS ของน้ำผสมสบู่ด้วย TDS Sensor

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 การทดลองการควบคุมอุปกรณ์แบบอัตโนมัติ

ในการทดลองนี้ผู้จัดทำได้ทำการเขียนคำสั่งการทำงานให้สามารถทำงานเปิดและปิดอุปกรณ์ต่างๆที่ใช้ในการควบคุมตู้ปลาได้อัตโนมัติตามค่าที่เราต้องการได้ โดยการทดลองจะเป็นการควบคุม Relay ให้เปิดหรือปิดเมื่อค่าจากเซ็นเซอร์มากกว่าหรือว่าน้อยกว่าค่าที่ผู้ใช้ได้ทำการตั้งไว้ โดยมีวิธีการทดลองดังนี้

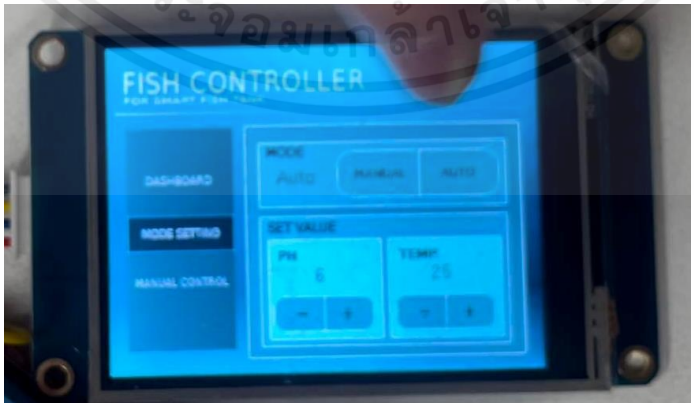
4.2.1 การทดลองตั้งค่าอุณภูมิ

ผู้ใช้จะทำการตั้งค่าอุณหภูมิที่จอสัมผัสไว้ที่ค่าใดค่าหนึ่ง และเมื่อค่าอุณหภูมิจาก Sensor มีค่ามากกว่าอุณหภูมิที่ตั้งไว้ ระบบจะทำการเปิด Relay ที่ควบคุมพัดลมให้ทำงาน และหากน้อยกว่า Relay จะทำการเปิด Relay ที่ควบคุมฮีทเตอร์ ให้ทำงานและปิด Relay ของพัดลม

```
////////////////////////////////////////////////////////////////////Auto Temperature/////////////////////////////////////////////////////////////////////////////////////////
                                                    //ค่า tempC คือค่าที่อ่านจาก sensor, ค่า tempset คือค่าที่ตั้งเอาไว้
  if(tempC == tempset){                             //เมื่อค่าที่ตั้งไว้มีค่าเท่ากับค่าที่อ่านมาจาก Sensor ก็จะทำการปิด Relay ที่ขา 18 และ 19 (ตัวที่ใช้ควบคุม พัดลม และ ฮีทเตอร์)

      digitalWrite(18,LOW);
      //delay(500);
      digitalWrite(19,LOW);
      //delay(500);

  }
  else if(tempC < tempset){                         //เมื่อค่าที่ตั้งไว้มีค่าน้อยกว่าค่าที่อ่านมาจาก Sensor ก็จะทำการเปิด Relay ที่ขา 19 (ตัวที่ใช้ควบคุม ฮีทเตอร์)    และเปิดขา 18 (ตัวที่ใช้ควบคุม พัดลม)
      digitalWrite(18,LOW);
      //delay(500);
      digitalWrite(19,HIGH);
      //delay(500);
  }
  else{                                             //เมื่อค่าที่ตั้งไว้มีค่ามากกว่าค่าที่อ่านมาจาก Sensor ก็จะทำการปิด Relay ที่ขา 19 (ตัวที่ใช้ควบคุม ฮีทเตอร์)    และเปิดขา 18 (ตัวที่ใช้ควบคุม พัดลม)
      digitalWrite(18,HIGH);
      //delay(500);
      digitalWrite(19,LOW);
      //delay(500);
  }
  }
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////rtc////////////////////
```

รูปที่ 4.15 โค้ดที่ใช้ควบคุมการเปิดหรือปิด Relay ของพัดลมและฮีทเตอร์

ในรูปที่ 4.14 ผู้ใช้ได้ทำการตั้งอุณหภูมิไว้ที่ 25 องศา

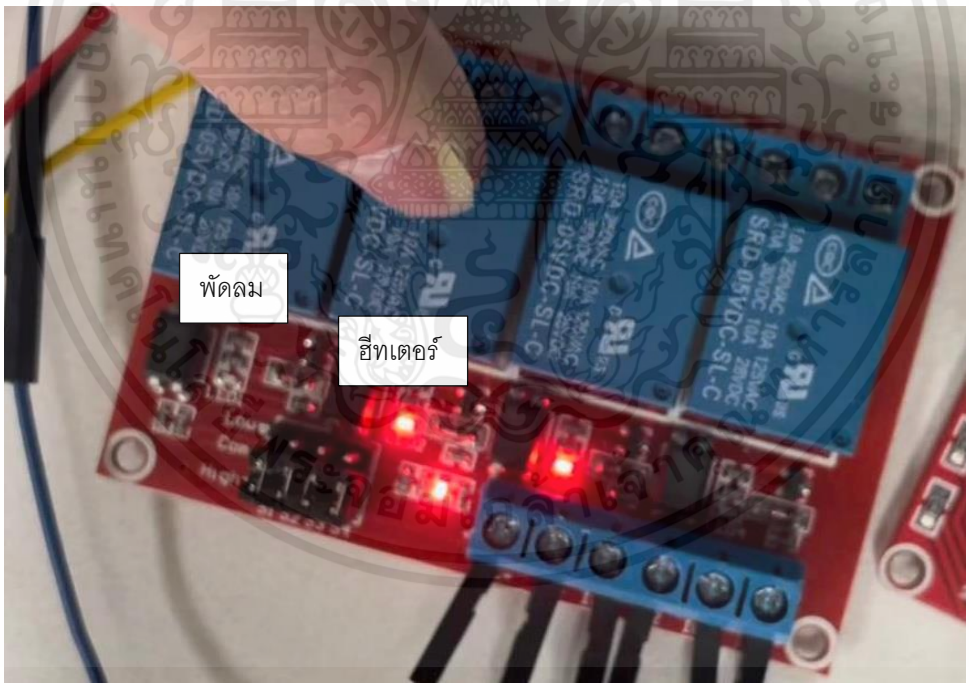


รูปที่ 4.16 ภาพของหน้าจอที่ปรับค่าอุณหภูมิไว้ที่ 25

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในรูปที่ 4.15 ค่าที่อ่านได้จาก Temperature Sensor มีค่าเท่ากับ 24.12 องศาเซลเซส ซึ่งมีค่าน้อยกว่าค่าที่ตั้งไว้คือ 25 องศาเซลเซส รีเลย์ตัวที่ควบคุมฮีทเตอร์จะทำงานดังรูปที่ 4.16

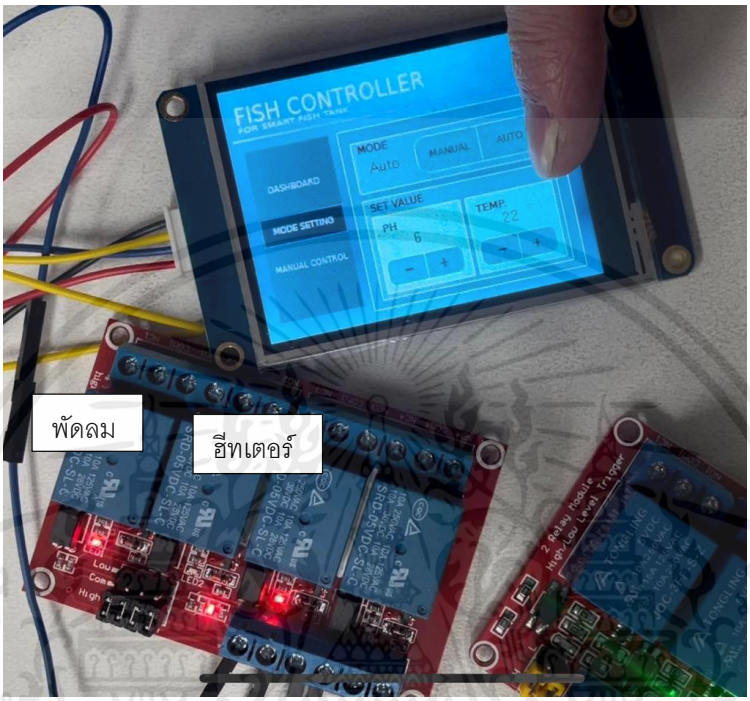
รูปที่ 4.17 รูปแสดงค่าได้จากเซ็นเซอร์



รูปที่ 4.18 ภาพขณะที่รีเลย์ตัวที่ควบคุมฮีทเตอร์ทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากนั้นผู้ใช้ได้ทำการเปลี่ยนค่าอุณหภูมิเป็น 22 องศาเซลเซียส ค่าจากเซ็นเซอร์จึงมีค่ามากกว่า รีเลย์จะทำการตัดสลับจาก Relay ของฮีทเตอร์ที่ทำงานอยู่ให้หยุดทำงาน และเปิด Relay ของพัดลมแทน ดังรูปที่ 4.17



รูปที่ 4.19 ภาพขณะที่รีเลย์ตัวที่ควบคุมพัดลมทำงาน

และในทำนองเดียวกันในส่วนของการควบคุม pH ก็จะใช้คำสั่งเดียวกัน ก็คือเมื่อ pH ที่ผู้ใช้ได้ตั้งไว้น้อยกว่าค่า pH จากเซ็นเซอร์ก็จะทำการเปิด Relay ที่ควบคุมปั๊ม pH down ถ้าหากมากกว่าก็จะเปิด pH up ให้ทำงาน และปิด pH down

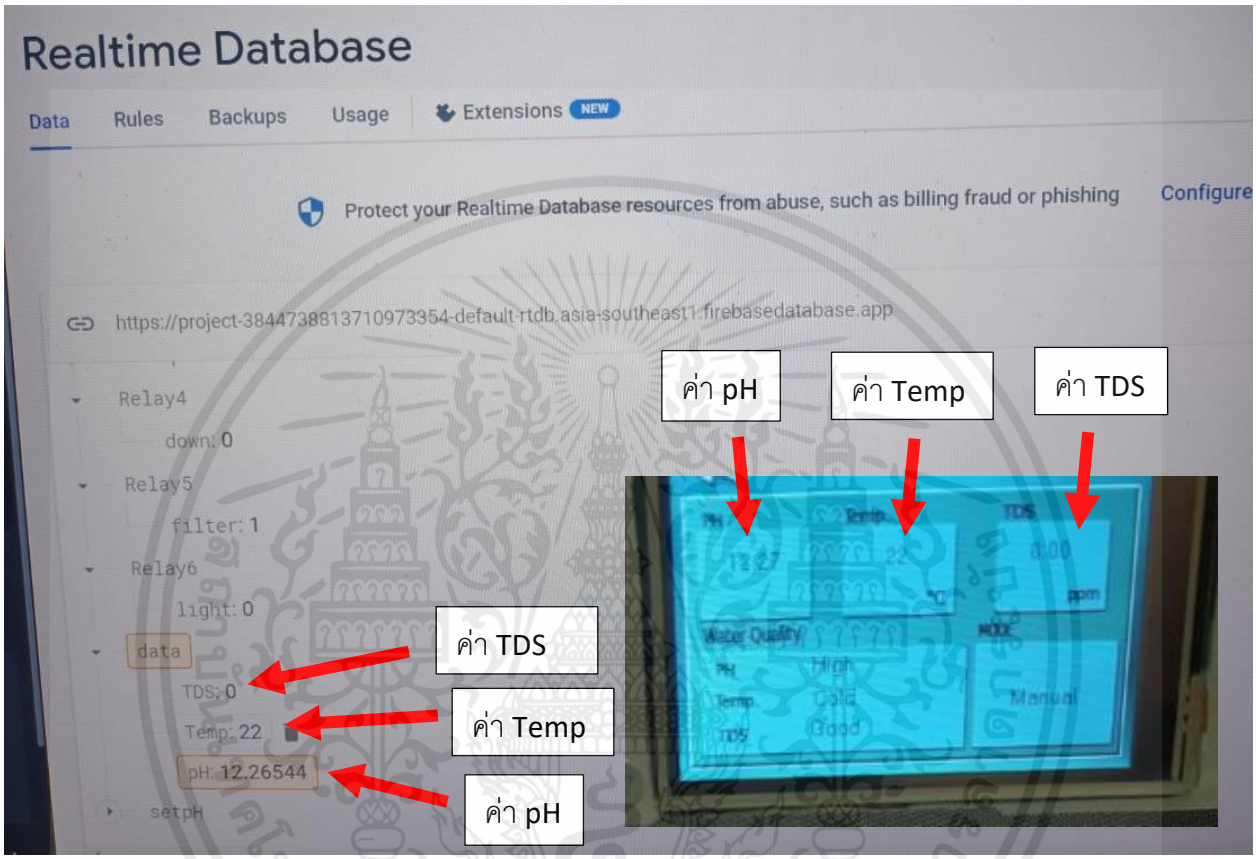
```
////////////////////////////////////////////////////////////////////Auto pH////////////////////////////////////////////////////////////////////
                                                                    //ค่า pHValue คือค่าที่อ่านจาก sensor, ค่า pHset คือค่าที่ตั้งเอาไว้
    if(pHValue > pHset + 1){
       digitalWrite(21,LOW);
       //delay(500);
       digitalWrite(23,HIGH);
       //delay(500);
}
  else if(pHValue < pHset - 1){
       digitalWrite(21,HIGH);
      // delay(500);
       digitalWrite(23,LOW);
       //delay(500);
  }
  else{
       digitalWrite(21,LOW);
       //delay(500);
       digitalWrite(23,LOW);
       //delay(500);
  }
```

รูปที่ 4.20 โค้ดที่ใช้ควบคุมการเปิดหรือปิด Relay ของ pH up และ pH down

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 การทดลองวัดค่าจากจอภาพเทียบกับ Firebase

ผู้จัดทำได้ทำการนำค่าเซ็นเซอร์ที่วัดได้มาแสดงค่าผ่านทางจอภาพ และนำค่าเซนเซอร์ที่วัดได้อัพขึ้น Firebase จากนั้นทำการเทียบค่าเซนเซอร์ที่วัดได้จากจอภาพและ Firebase ได้ผลดังนี้



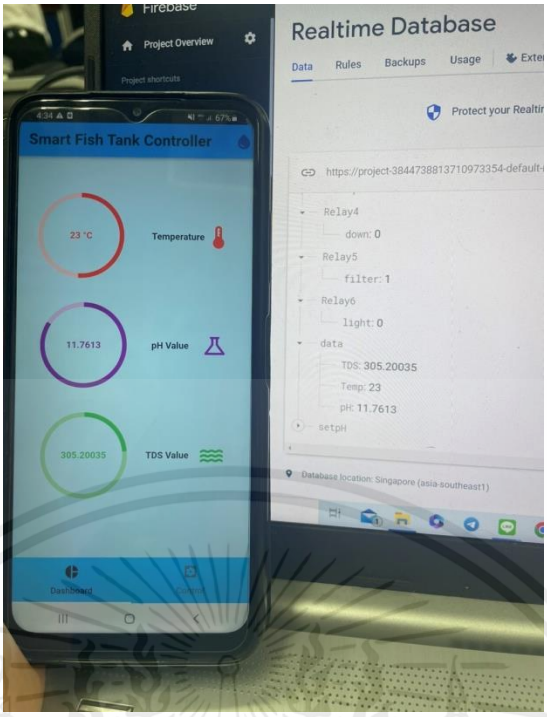
รูปที่ 4.21 ภาพการเทียบค่าเซนเซอร์ที่วัดได้จากจอภาพและ Firebase

4.4 การทดลองวัดค่าจากแอพลิเคชั่นเทียบกับ Firebase

4.4.1 การทดลองวัดค่าจาก Sensor

ผู้จัดทำได้ทำการนำค่าเซ็นเซอร์ที่วัดได้มาแสดงค่าผ่านทางจอภาพ และนำค่าเซนเซอร์ที่วัดได้อัพขึ้น Firebase จากนั้นนำค่าเซนเซอร์ที่วัดได้จาก Firebase มาออกแอพลิเคชั่น ได้ผลดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

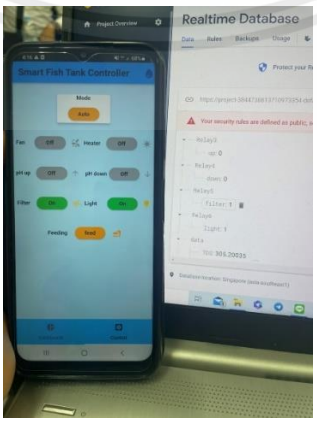


รูปที่ 4.22 ภาพการเทียบค่าเซนเซอร์ที่วัดได้จากแอพลิเคชั่นและ Firebase

4.4.2 การทดลองการควบคุม Relay ผ่านแอพลิเคชั่น

ผู้จัดทำได้ทำการออกแบบการควบคุม Relay ผ่านแอพลิเคชั่น โดยเมื่อกดสั่งให้ Relay ทำงานผ่านแอพลิเคชั่น หรือกดให้อุปกรณ์ “on” แอพลิเคชั่นจะส่งค่าตัวเลข 1 ไปยัง Firebase และถ้ากดสั่งให้ Relay หยุดการทำงานผ่านแอพลิเคชั่น หรือกดให้อุปกรณ์ “off” แอพลิเคชั่นจะส่งค่าตัวเลข 0 ไปยัง Firebase ได้ผลดังนี้

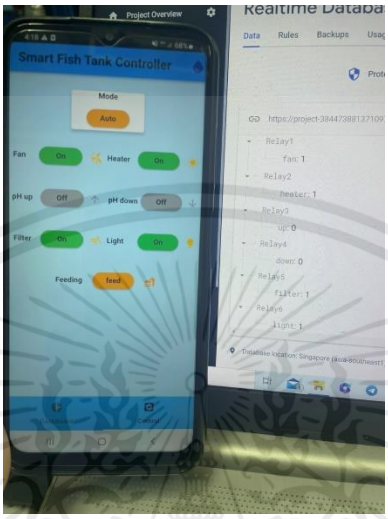
จากรูปที่ 4. ทำการสั่งให้ Filter และ Light ทำงานในแอพลิเคชั่น ค่าใน Firebase ที่เป็นส่วนของ Filter และ Light ซึ่งมี Path เป็น Relay5/filter กับ Relay6/light มีค่าเป็น 1 ส่วนอุปกรณ์อื่นๆไม่ได้สั่งให้ทำงานจึงมีค่าเป็น 0



รูปที่ 4.23 สั่งผ่านแอพลิเคชั่นให้ Filter และ Light ทำงาน

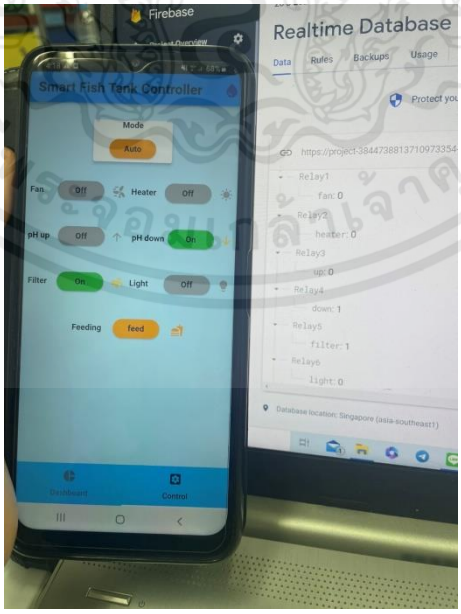
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 4. ทำการสั่งให้ Fan และ Heater ทำงานเพิ่ม ค่าใน Firebase ที่เป็นส่วนของ Fan และ Heater ซึ่งมี Path เป็น Relay1/fan กับ Relay2/heater มีค่าเป็น 1 สำหรับ Filter และ Light ไม่ได้สั่งให้หยุดทำงาน ใน Firebase จึงยังมีค่าเป็น 1 อยู่ ส่วนอุปกรณ์อื่นๆไม่ได้สั่งให้ทำงานจึงมีค่าเป็น 0



รูปที่ 4.24 สั่งให้ Fan และ Heater ทำงานเพิ่ม

จากรูปที่ 4. ทำการสั่งให้ pH down ทำงาน ค่าใน Firebase ที่เป็นส่วนของ pH down ซึ่งมี Path เป็น Relay4/down มีค่าเป็น 1 สำหรับ Filter ไม่ได้สั่งให้หยุดทำงาน ใน Firebase จึงยังมีค่าเป็น 1 อยู่ ส่วนอุปกรณ์อื่นๆไม่ได้สั่งให้ทำงานจึงมีค่าเป็น 0



รูปที่ 4.25 สั่งให้ pH down และ Filter ทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปผลและข้อเสนอแนะ

5.1 สรุปผล

จากการทดลองการทำงานของระบบตู้ปลาอัจฉริยะพบว่าค่าพารามิเตอร์ที่ได้จากเซนเซอร์ยังคงมีความคลาดเคลื่อนอยู่ โดยเฉพาะ pH sensor ที่มีความไวต่อกรดมาก ด้านการควบคุมการทำงานของระบบตู้ปลาอัจฉริยะ สามารถนำค่าพารามิเตอร์ต่างๆที่ได้จาก Temp, pH, TDS sensors ออกหน้า Dashboard ของจอ Nextion ได้ สามารถใช้คำสั่งควบคุมการทำงานของรีเลย์และสามารถสั่งการผ่านจอ Nextion ได้ โดยโหมดการทำงานของระบบตู้ปลาอัจฉริยะนี้มีอยู่ 2 โหมด คือ โหมด Manual ที่สามารถสั่งเปิด/ปิดการทำงานของอุปกรณ์เลี้ยงปลาผ่านรีเลย์ได้เลย และโหมด Auto ที่อุปกรณ์จะทำงานเมื่อค่าพารามิเตอร์สำหรับการเลี้ยงปลาที่เรากำหนดไว้มีค่าต่ำ/สูงกว่าเกณฑ์ที่เรากำหนด ซึ่งทำการทดสอบแล้วพบว่า 2 โหมดทำงานได้อย่างมีประสิทธิภาพ และยังสามารถแสดงค่าจาก Sensor และควบคุมอุปกรณ์ต่างๆได้ผ่านแอพพลิเคชั่นมือถือได้ผ่านระบบอินเตอร์เน็ต

5.2 ข้อเสนอแนะ

1. สำหรับ pH sensor ยังมีค่าความคลาดเคลื่อนสูง และพบว่า pH sensor ตัวนี้มีความไวต่อน้ำที่มีฤทธิ์เป็นกรด ควรจะต้องทำการปรับแต่งวงจรให้มีความแม่นยำขึ้น หรือไปใช้ pH sensor ที่มีความแม่นยำมากขึ้น

2. ในฟังก์ชั่นการทำงาน ควรมีระบบเปลี่ยนถ่ายน้ำแบบอัตโนมัติโดยมีปั๊มที่ใช้ถ่ายน้ำเก่าออก และปั๊มที่นำน้ำใหม่เข้ามา

3. ในการเลี้ยงปลาค่าของน้ำที่ควรคำนึงถึงนั้นมีอีกมาก เช่น ความเค็ม, ไนโตรเจนในน้ำ, คาร์บอน ถ้าสามารถเพิ่ม Sensor ที่ใช้วัดค่าเหล่านี้ได้ และสามารถควบคุมอุปกรณ์ที่สามารถปรับแต่งค่าเหล่านี้ได้ ระบบก็จะสมบูรณ์แบบมากขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

[1] ใบความรู้ที่ 1.3 เรื่องหน่วยประมวลผลกลาง (central processing unit), สืบค้นเมื่อ 19 พฤศจิกายน 2565. จาก.https://docs.google.com/document/preview?hgd=1&Id=1z5k9_puMOsmf2smblO1wS4WNNRB- sa4zPq3E03v9YIY

[2] ใบความรู้ที่ 1.4 เรื่องหน่วยความจำ, สืบค้นเมื่อ 19 พฤศจิกายน 2565. จาก.https://docs.Google.com/document/preview?hgd=1&id=1HcAcgHY10oPlaxIn9u0K5Dxbi6F1ClgVq06j1O437EE

[3] ไมโครคอนโทรลเลอร์ - วิกิพีเดีย(wikipedia.org), สืบค้น เมื่อ 19 พฤศจิกายน 2565. จาก. https://th.wikipedia.org/wiki/%E0%B9%84%E0%B8%A1%E0%B9%82%E0%B8%84%E0%B8%A3%E0%B8%84%E0%B8%AD%E0%B8%99%E0%B9%82%E0%B8%97%E0%B8%A3%E0%B8 %A5%E0%B9% 80%E0%B8%A5%E0%B8%AD%E0%B8%A3%E0%B9%8C

[4] 07_ch2.pdf (siam.edu), สืบค้นเมื่อ 19 พฤศจิกายน 2565. จาก. https://researchsystem.siam.edu /images/EE/ProjectEE2/wya1/07_ch2.pdf

[5] ไมโครโปรเซสเซอร์กับไมโครคอนโทรลเลอร์: อะไรคือ ความแตกต่าง? – บริษัท ชุนหลงเหว่ย จำกัด (shunlongwei.com), สืบค้นเมื่อ 20 พฤศจิกายน 2565. จาก. https://www.shunlongwei.com/th/microprocessor-vs-microcontroller-what-is-the-difference/

[6] บทความ – ArtronShop บอร์ดอิเล็กทรอนิกส์ Arduino ESP32 ESP8266 : Inspired by LnwShop.com, สืบค้นเมื่อ 20 พฤศจิกายน 2565.จาก. https://www.artronshop.co.th/article

[7] รู้จักกับโปรแกรม Arduino IDE – futurekit, สืบค้นเมื่อ 20 พฤศจิกายน 2565. จาก. https://www.futurekit.com/th/content/10793/%E0%B8%A3%E0%B8%B9%E0%B9%89%E0%B8%88%E0%B8%B1%E0%B8%81%E0%B8%81%E0%B8%B1%E0%B8%9A%E0%B9%82%E0%B8%9B%E0%B8%A3%E0%B9%81%E0%B8%81%E0%B8%A3%E0%B8%A1-arduino-ide

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

[8] รู้จักค่า ph คือความเป็นกรด-ด่างของสารละลาย ของเหลวและน้ำ (neonics.co.th), สืบค้นเมื่อ 20 พฤศจิกายน 2565.จาก. https://www.neonics.co.th/ph/what-is-ph-and-measurement.html

[9] tds คืออะไร (neonics.co.th), สืบค้นเมื่อ 20 พฤศจิกายน 2565.

จาก.https://www.neonics.co.th/water-quality-testing/tds.html

[10] การติดต่อกับ Nextion HMI.pdf – Google ไดรฟ์, สืบค้นเมื่อ 21 พฤศจิกายน 2565.จาก. https://drive.google.com/file/d/1sSybmd8Hatf9tgup63WpAEmqJS5RL7ip/view

[11] น้ำสำหรับปลาสวยงาม, สืบค้นเมื่อ 5 พฤษภาคม 2566 จาก https://home.kku.ac.th/pracha/Water.htm

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Table 1: ESP32-WROOM-32 Specifications

Categories	Items	Specifications
Certification	RF certification	See certificates for ESP32-WROOM-32
	Wi-Fi certification	Wi-Fi Alliance
	Bluetooth certification	BQB
	Green certification	RoHS/REACH
Test	Reliablity	HTOL/HTSL/uHAST/TCT/ESD
Wi-Fi	Protocols	802.11 b/g/n (802.11n up to 150 Mbps)
		A-MPDU and A-MSDU aggregation and 0.4 μs guard interval support
	Center frequency range of operating channel	2412 ~ 2484 MHz
Bluetooth	Protocols	Bluetooth v4.2 BR/EDR and Bluetooth LE specification
	Radio	NZIF receiver with –97 dBm sensitivity
		Class-1, class-2 and class-3 transmitter
		AFH

Categories	Items	Specifications
	Audio	CVSD and SBC
Hardware	Module interfaces	SD card, UART, SPI, SDIO, I2C, LED PWM, Motor PWM, I2S, IR, pulse counter, GPIO, capacitive touch sensor, ADC, DAC, Two-Wire Automotive Interface (TWAI®), compatible with ISO11898-1 (CAN Specification 2.0)
	On-chip sensor	Hall sensor
	Integrated crystal	40 MHz crystal
	Integrated SPI flash	4 MB
	Operating voltage/Power supply	3.0 V ~ 3.6 V
	Operating current	Average: 80 mA
	Minimum current delivered by power supply	500 mA
	Recommended operating ambient temperature range	–40 °C ~ +85 °C
	Package size	18 mm × 25.5 mm × 3.10 mm
	Moisture sensitivity level (MSL)	Level 3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PRELIMINARY

DALLAS SEMICONDUCTOR

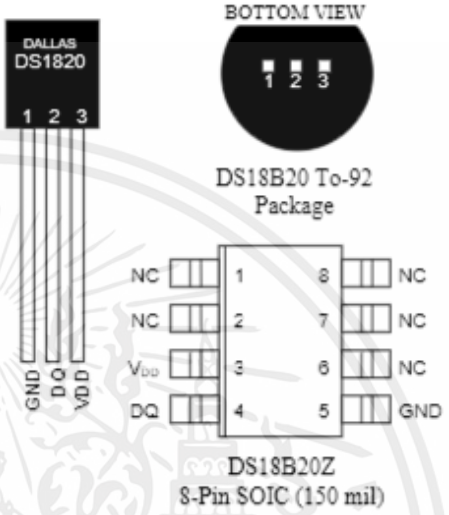
DS18B20
Programmable Resolution 1-Wire® Digital Thermometer

www.dalsemi.com

FEATURES

- Unique 1-Wire interface requires only one port pin for communication
- Multidrop capability simplifies distributed temperature sensing applications
- Requires no external components
- Can be powered from data line. Power supply range is 3.0V to 5.5V
- Zero standby power required
- Measures temperatures from -55°C to +125°C. Fahrenheit equivalent is -67°F to +257°F
- ±0.5°C accuracy from -10°C to +85°C
- Thermometer resolution is programmable from 9 to 12 bits
- Converts 12-bit temperature to digital word in 750 ms (max.)
- User-definable, nonvolatile temperature alarm settings
- Alarm search command identifies and addresses devices whose temperature is outside of programmed limits (temperature alarm condition)
- Applications include thermostatic controls, industrial systems, consumer products, thermometers, or any thermally sensitive system

PIN ASSIGNMENT



PIN DESCRIPTION

GND - Ground
DQ - Data In/Out
V_{DD} - Power Supply Voltage
NC - No Connect

DESCRIPTION

The DS18B20 Digital Thermometer provides 9 to 12-bit (configurable) temperature readings which indicate the temperature of the device.

Information is sent to/from the DS18B20 over a 1-Wire interface, so that only one wire (and ground) needs to be connected from a central microprocessor to a DS18B20. Power for reading, writing, and performing temperature conversions can be derived from the data line itself with no need for an external power source.

Because each DS18B20 contains a unique silicon serial number, multiple DS18B20s can exist on the same 1-Wire bus. This allows for placing temperature sensors in many different places. Applications where this feature is useful include HVAC environmental controls, sensing temperatures inside buildings, equipment or machinery, and process monitoring and control.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PH Sensor E-201-C

Technical Manual Rev 1r0

The pH sensor Module consist of PH sensor also called as PH probe and a Signal Conditioning Board which gives an output which is proportional to the PH Value and can be interfaced directly to any Micro-controller.
The pH sensor components are usually combined into one device called a combination pH electrode. The measuring electrode is usually glass and quite fragile. Recent developments have replaced the glass with more durable solid-state sensors. The preamplifier is a signal-conditioning device. It takes the high-impedance pH electrode signal and changes it into a low impedance signal which the analyzer or transmitter can accept. The preamplifier also strengthens and stabilizes the signal, making it less susceptible to electrical noise.

pH and ORP probes are both used for measuring the acidic intensity of liquid solutions. A pH probe measures acidity on a scale from 0 to 14, with 0 being the most acidic and 14 being the most basic. Similarly, an Oxidation-Reduction Potential (ORP) probe returns a voltage proportional to the tendency of the solution to gain or lose electrons from other substances (which is linked directly to the pH a substance).

Applications:

- Can be used as different tester, pH meter or controller
- Suitable for most aquariums, hydroponics, laboratory etc

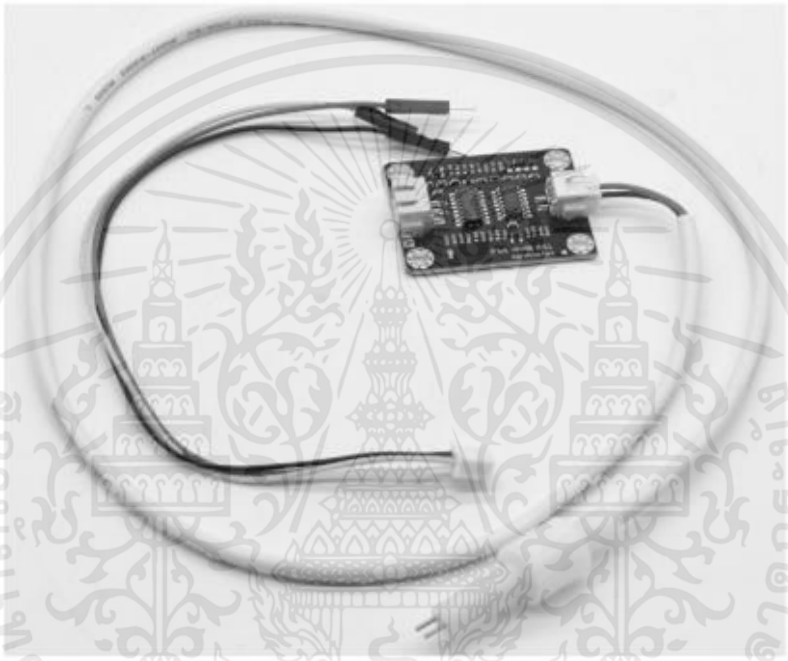
General Specifications:

Input supply voltage: 5V
Working current: 5 – 10mA
Detection concentration range: PH 0 – 14
Detection range of temperature: 0 – 80 degC
Response Time:: ≤ 5S
Stability Time: ≤ 60S
Output: Analog
Power Consumption: ≤ 0.5W
Working Temperature: -10 to +50 deg C
Working Humidity: 95%RH (nominal humidity 65%RH)
Weight: 25g
PCB Dimension: 42mm x 32mm x 20mm

©2017 by e-Gizmo Mechatronix Central

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

keyestudio TDS Meter V1.0



Description

Keyestudio TDS sensor kit is compatible with Arduino controllers, plug and play, easy to use.

It can be applied to measure TDS value of the water, to reflect the cleanliness of the water.

TDS (Total Dissolved Solids) indicates that how many milligrams of soluble solids dissolved in one liter of water. In general, the higher the TDS value, the more

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

soluble solids dissolved in water, and the less clean the water is.

Therefore, the TDS value can be used as one of the references for reflecting the cleanliness of water.

Measuring the TDS value in the water is to measure the total amount of various organic or inorganic substances dissolved in water, in the unit of ppm or milligrams per liter (mg/l).

Its Electrode can measure conductive materials, such as suspended solids, heavy metals and conductive ions in water.

The module comes with four 3.2mm fixed holes, easy to mount on any other devices.

Technical Parameters

TDS Meter:

- Input Voltage: DC 3.3 ~ 5.5V
- Output Voltage: 0 ~ 2.3V
- Working Current: 3 ~ 6mA
- TDS Measurement Range: 0 ~ 1000ppm
- TDS Measurement Accuracy: ± 10% F.S. (25 °C)
- Module Interface: XH2.54-3P
- Electrode Interface: XH2.54-2P

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

NX3224T028

Overview
Nextion Models
Specifications
Electronic Characteristics
Working Environment & Reliability Parameter
Interfaces Performance
Memory Features
Product Dimensions

Overview

Nextion is a seamless Human Machine Interface (HMI) solution that provides a control and visualisation interface between a human and a process, machine, application or appliance. Nextion is mainly applied to IoT or consumer electronics field. It is the best solution to replace the traditional LCD and LED Nixie tube. With the Nextion Editor software (Official Download), users are able to create and design their own interfaces for Nextion display.

Package includes: Nextion Display, connecting wire, a power supply test board.
Go Shopping: NX3224T028 (IM150416004)
Note: the small power supply test board and connecting wire inside the package allow you to test if the electrical supply is enough or not. See the image below on how to use it.

Specifications

	Data	Description
Color	64K 65536 colors	16 bit 565, 5R-6G-5B
Layout size	85(L)×49.8(W)×4.6(H)	NX3224T028_011N
	85(L)×49.8(W)×5.8(H)	NX3224T028_011R
Active Area (A.A.)	70.2mm(L)×49.8mm(W)	
Visual Area (V.A.)	57.6mm(L)×43.2mm(W)	
Resolution	320×240 pixel	Also can be set as 240×320
Touch type	Resistive	
Touches	> 1 million	
Backlight	LED	
Backlight lifetime (Average)	>30,000 Hours	
Brightness	200nit (NX3224T028_011N)	0% to 100%, the interval of adjustment is 1%
	180 nit (NX3224T028_011R)	0% to 100%, the interval of adjustment is 1%
Weight	32g (NX3224T028_011N)	
	29.5g (NX3224T028_011R)	

Electronic Characteristics

	Test Conditions	Min	Typical	Max	Unit
Operating Voltage		4.75	5	7	V
Operating Current	VCC=+5V, Brightness is 100%	–	65	–	mA
	SLEEP Mode	–	15	–	mA
Power supply recommend: **5V, 500mA, DC**					

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้