# การใช้งานกับ Blynk

ในบนนี้แนะนำการใช้งาน Blynk ซึ่งเป็นโปรแกรมรับแสดงผลข้อมูลจากอุปกรณ์ IoT ต่างๆ เราจะเริ่มจากการทำ ทดสอบอุปกรณ์ IoT ก่อน ซึ่งใช้บอร์ดพัฒนาเป็น ESP32 หรือ M5Stack และ ESP8266 หรือ Node MCU ด้วยการทดสอบ การใช้งาน WiFi และเซนเซอร์ DHT22 เมื่ออุปกรณ์ต่างๆ พร้อมแล้วก็ทดสอบส่งข้อมูลเพื่อการแสดงผลกับ Blynk ต่อไป

สำหรับ บอร์ด Arduino ไม่มีระบบ WiFi ในตัว จะต้องใช้ระบบ WiFi ตัวอื่นช่วย เช่น ESP32 หรือ ESP8266 ด้วย สองตัวนี้ก็สามารถส่งผ่านสัญญานต่อไปยัง WiFi ได้

# ESP32 ต่อกับ WiFi

การทำงานกับ WiFi ของ ESP32 มีอยู่ด้วยกัน ด้วยโหมด STA (Station Accession Point) เป็นการติดต่อกับ ตัว ปล่อยสัญญาณ สำหรับ ESP32 หรือ M5Stack ต่อเชื่อมโดยทำหน้าที่เป็นเครื่องลูกข่าย

ใช้โหมด STA รติดต่อผ่านการใช้ฟังก์ชัน begin(name, password) เมื่อต่อเชื่อมได้จะมีสถานะ โดยใช้ฟังก์ชัน status( ) โดยเทียบกับ สถานะ WL\_CONNECTED ดังตัวอย่างต่อไปนี้

Code 1. ESP32 ต่อเชื่อม WiFi ในโหมด STA

```
#include <WiFi.h>
```

```
const char* WIFI STA NAME = "OPPO A3s";
const char* WIFI_STA_PASS = "12345678";
void setup() {
  Serial.begin(115200);
  WiFi.mode(WIFI_STA);
  WiFi.begin(WIFI STA NAME, WIFI STA PASS);
  while(WiFi.status() != WL CONNECTED){
    delay(500);
    Serial.print(".");
  }
  Serial.println();
  Serial.println("WiFi Connected!");
  Serial.print("IP:");
  Serial.println(WiFi.localIP());//Router gives IP.
}
void loop() {
  // put your main code here, to run repeatedly:
}
```



**รูป** 1 ผลการต่อเชื่อม ในโหมด STA

#### ESP8266 (Node MCU) ต่อกับ WiFi

Node MCU ใช้ ไลบรารี ESP8266WiFil.h ในการต่อเชื่อมกับ WiFi ในการต่อเชื่อมก็ใช้ SSID กับ รหัสผ่านกับฟังก์ชัน WiFi.begin(SSID, PASSWORD) ใช้ while ต่อเชื่อมจนกว่าจะต่อเชื่อมได้

Code 2. Node MCU การต่อเชื่อม WiFi

```
#include <ESP8266WiFi.h>
const char* SSID = "OPPO A3s";
const char* PASSWORD = "12345678";
void setup() {
    Serial.begin(115200);
    delay(10);
    Serial.println();
    Serial.print("Connecting to ");
    Serial.println(SSID);
    //Connect to WiFi network
    WiFi.begin(SSID, PASSWORD);
    while(WiFi.status()!= WL_CONNECTED){
      delay(500);
      Serial.print(".");
    }
    Serial.println("");
    Serial.println("WiFi connected");
    Serial.println("IP address: ");
    Serial.println(WiFi.localIP());
}
void loop() {
  // put your main code here, to run repeatedly:
```

```
}
```

© COM7	_	×
		Send
		~
WiFi connected		
IP address:		
192.168.43.222		

**รูป 2** ผลการต่อเชื่อม WiFi

### ทดสอบเซนเซอร์วัดอุณหภูมิ (DHT22)

เพื่อการเป็นการทดสอบข้อมูลส่งไปทำงานกับ Blynk จะเลือกใช้เซนเซอร์วัดอุณหภูมิ DTT22 ซึ่งถ้าใครมี DTT11 ก็ ใช้แทนกันได้

ก่อนอื่นต้องนำเข้าไลบรารี การทำงานกับเซนเซอร์นี้เสียก่อน ซึ่งมีอยู่หลายตัว ในที่นี้ะจะแนะนำ DHTesp ซึ่งเป็น ไลบรารี่ที่ใช้งานง่ายตัวหนึ่ง การติดตั้งให้เข้าไปที่รายการ Sketch> Include Library > Manage Libraries... ตาม **รูป 3** 

	ESP32   Arduino 1.8.12	825		×			
File Edit	Sketch Tools Help						
00	Verify/Compile	Ctrl+R			0		
DHT_E	Upload Upload Using Programmer	Ctrl+U Ctrl+Shift+U		5	2		
#includ	Export compiled Binary	Ctrl+Alt+S			^		
FIREIGO	Show Sketch Folder	Ctrl+K		Δ.			
#ifndef	Include Library	Include Library					
<pre>#pragma #error</pre>	Add File	Add File					

**รูป 3** เลือก Mange Libraries

หลังจากนั้นจะมีหน้าต่าง Library Manger ตาม**รูป 4** ให้ค้นหา Dht เพื่อแสดงรายการไลบรารี่ที่เฎี่ยวกับคำค้นฎี่มา ทั้งหมด ต่อมาให้เลื่อนหา Arduino ESP library for DHT11, DHT22 และติดตั้งไลบรารี่นี้

💿 Li	ibrary Man	ager									
Туре	All	~	Topic	All	~	Dht					
by Ar ch <u>Me</u> Se	beegee_ duino ESF anges: Re ore info elect version	tokyo V Plibrary educe C	for DI PU usa Install	1.17.0 INS HT11, DHT2 ige and add	TALLED 2, etc Temp d decimal p	& Humidity Sense art for DHT11	ors Optimize	ed libray to r	natch ESP32	erequirement	s. Last

รูป 4 เลือก Arduino ESP library

เมื่อติดตั้งไลบรารี่ Arduino ESP เสร็จแล้ว ทดสอบการอ่านอุณหภูมิ จากตัวอย่างโปรแกรม โดยเลือก รายการที่ File > Examples > DHT sensor library for ESPx > DTH\_ESP32 กรณีใช้บอร์ดพัฒนาเป็น ESP32 หรือ M5Stack แต่ถ้าใช้ บอร์ด Node MCU หรือ ESP8266 ก็เลือก เป็น DHT ESP8266

เมื่อเลือกตัวอย่างโปรแกรมแล้ว จะมีค่า กำหนดชนิดเซนเซอร์เป็นค่า DTH11 แต่ถ้าใช้ตัวเซนเซอร์ทดสอบเป็น DHT22 ก็ให้แก้ไขให้ตรงกับเซนเซอร์ที่ใช้

นอกจากนี้ โปแกรมนี้กำหนดขา (Pin) ต่อเชื่อมกับขาสัญญานเซนซอร์ที่ 17 แสดงว่าต้องต่อสายสัญญานวัดค่าที่ขา 17 (GPIO 17) ของบอร์ด ESP32 จาก**รูป 5** แสดงการต่อ ESP32 กับ DHT ซึ่งในรูปนี้ไม่มีต่อกับตัวต้านทานไว้ด้วย เพราะ โดยทั่วไปเซนเซอร์จะมีระบบป้องกันอยู่แล้ว ไม่จำเป็นต้องต่อกับตัวต้านแทนไว้ก็ได้ แต่ถ้าไม่มีระบบป้องกันก็ให้ใส่ตัวต้านทาน 220 โอมห์ ขั้นระหว่าง ขาสัญญาณ (Sig) กับ ขา GPIO 17



**รูป 5.1** ESP32 ต่อกับเซนเซอร์ เป็น DHT22/DHT11



**รูป 5.2** M5Stack ต่อกับเซนเซอร์ เป็น DHT22/DHT11



**รูป 6** เลือก ชนิดเซนเซอร์ เป็น DHT22

สุดท้ายกดรันโปรแกรม จะได้ผลการวัดค่าผ่าน หน้าต่าง Tool > Serial Monitor ดังผลรูปต่อไปนี้ เป็นอันเสร็จสิ้น การทดสอบเซนเซอร์

6	20 COM7	_	$\times$
			Send
			~
DF	HT ESP32 example with tasks		
DF	HT initiated		
te	empTask loop started		
1	F:31.50 H:67.10 I:38.03 D:24.67 Comfort_TooHot		
1	F:31.60 H:73.40 I:40.44 D:26.27 Comfort_HotAndHumid		

รูป 7 ผลการทำงานอ่านอุณหภูมิ และความชื้น

# ติดตั้ง Blynk บนโทรศัพท์เคลื่อน

ให้เลือกแอปพลิเคซัน Blynk เช่น จาก PlayStore ของ Android ติดตั้งแอปฯ นี้ ในขั้นตอนติดตั้งต้องกรอกข้อมูล email เพื่อการส่งรหัสการต่อเชื่อม เมื่อติดตั้งเสร็จ ให้เปิดแอปฯ นี้จะมีขั้นตอนการสร้างแอปฯ เชื่อมกับเซนเซอฐ์ต่อไป โดย สมมุติว่าเชื่อมกับ เซนเซอร์อุณหภูมิที่ได้ทดลองก่อนหน้านี้ ดังมีลำดับคือ

- 1. เลือก New Project (**รูป 8** ซ้าย)
- เลือก ตั้งชื่อ (Project Name) เช่น Temperature-Humidity, บอร์ดที่ใช้พัฒนา (CHOOSE DEVICE) เช่น ใช้กับบอร์ด ESP32 Dev Board, เลือก การต่อเชื่อม (CONNECTION TYPE) เช่น WiFi, เลือกแบบการแสดง เช่น DARK, เสร็จแล้ว กด Create (รูป 8 กลาง)
- 3. เมื่อกดสร้างแล้ว แอปฯ ที่สร้างก็จะส่งรหัส (Auth Token) การต่อเชื่อมไปยังอีเมล์ (**รูป 8** ขวา)

CPH1803  C  CPH1803  C   C  CPH1803  C   C   C   C   C   C   C   C   C	– 🗆 X	– 🗆 X	– 🗆 X
Image: service control       Image: service control         Image: service control       Image: service contro         Image: service contr	CPH1803 🖓 🌣 🖬 🙆 🔹 🕄 C 🙂	CPH1803 🙃 🌣 🝽 🙆 🕶 🕄 C 🙂	CPH1803 🖓 🌣 🛤 🔯 🔹 🕫 C 🙂
New Project     New Project     Wi-Fi        Wi-Fi <th>+</th> <th>Project Name</th> <th></th>	+	Project Name	
My Apps     Ark     Dark     Dark     Dark     Light     OK     Ok     Ok     Don't show again     Create	New Project	ESP32 Dev Board $\downarrow$	15 <sup>0</sup> ."
My Apps ARK LIGHT Create		CONNECTION TYPE Wi-Fi	Auth Token was sent to: theerapol.lim@gmail.com
My Apps PARK LIGHT OK Don't show again PAR Community Create	660	DHEME	You can also find it in @Project Settings
PR Community Create	My Apps	DARK LIGHT	ОК
Community			Don't show again
	<sup>چQ</sup> و Community	Create	
	< 0 ≡		< 0 ≡

รูป 8 ซ้าย เมื่อเริ่มสร้าง Project, กลาง กำหนดของแอปฯ, ขวา สำเร็จและส่งรหัสผ่านอีเมล์

- 4. ผลการสร้างแอปฯ ตอนนี้ยังไม่มีภาพหรือกราฟฟิกวัดค่าใดๆ ให้คลิกที่เครื่องหมาย บวก (+) ด้านบน (**รูป 9** ซ้าย)
- 5. จะได้หน้าต่าง Widget Box เพื่อเลือกกราฟฟิกการวัดต่างๆ ในที่นี้ให้เลือก Gauge (**รูป 9** กลาง)
- 6. ผลการเลือก Gauge จะปรากฏ กราฟฟิก เป็นเหมือนหน้าปัดวัดค่า (**รูป 9** ชวา)

12:35	36 M W		0.4.00.4		@ # <u>38</u> 04		samu	รดสาวแต่ว		Ist luburin 1.4	7 GB	sanal	คสานตัว		14 lunar	47 GB
O	Temperature-I	$\bigcirc$	$\oplus$	$\triangleright$	$\triangleleft$		Widget E	Box		Θ	Temperature-I	Ø	$\oplus$	$\triangleright$		
							PGY BALANCE 1 <b>± 1,500</b>	+ Ac	id	GAUGE	$\sim$		11 - 11 - 11 21 - 12 - 12 - 12 31 - 12 - 12 - 12			
							Step V +500		Ē	ſ						
										•						
							Value Displa	ау	(i)							
						हर	Labeled Val	ue								
						<del>A</del>	LED +100									
							Gauge		(i)							
							LCD +400		I.							
						ulti	SuperChart		i							
							Terminal			8-1						
			$\triangleleft$				0	$\triangleleft$					$\triangleleft$			

ร**ูป 9** ซ้าย Project ที่สร้างให้, กลาง เลือกการวัด Gauge, ขวา ผลการเลือก Gauge

- 7. ให้เลือก Gauge ของ**รูป 9** ขวา จะได้ผลการกำหนดค่าต่างๆ ของ Gauge (**รูป 10** ซ้าย)
- คลิก PIN จะได้เลือก Virtual PIN ให้เลือก V1 ซึ่งจะแทน PIN ของ Blynk เอง ไม่ใช้ PIN ในบรอร์ดพัฒนา ESP32 หรือ ESP8266 (รูป 10 กลาง) แล้วกด OK
- เลือกชื่อ Gauge เช่น Temperature หากไม่เปลี่ยนอะไร ก็จะมีชื่อว่า Gauge, เลือกค่าต่าสุด-สูงสุด 0 -100 และ แสดงผล 2 ต่ำแหน่ง แทนด้วยเครื่องหมาย ##, และเลือกระยะถี่ในการอ่านครั้ง 2 วินาที (2 sec) (รูป 10 ขวา) ถือเป็น อันเสร็จสิ้น สำหรับตัววัดที่ 1 ซึ่งใช้วัดอุณหภูมิ
- 10. ต่อไป สร้างตัววัดอีกตัว ใช้วัด ความชื้น ก็ทำในลักษณะเดียวกับตัววัดอุณหภูมิ แต่เลือกใช้ Virtual PIN เป็น V2

12:45 10.06	(i) + (iii)+	สอกสปอกสานสำ	ិទាំងលោ (	/47/68	anne the standing of	6	Things of a top
← Gauge Settings	i.	← Gauge Set			← Gauge	Settings	i
					Gauge		
Gauge					PIN	0	100
PIN 0	1023	PIN 0	102	23	LABEL /pin.##/		
LABEL		Select pin		OK			
e.g: Temp: /pin/ ºC					Ттт		
DESIGN			PIN				
		1040	VO			2 sec ↓	
READING RATE	=	virtual	V1 V2				
1 sec $\downarrow$			<b>V3</b> V4			🖹 Delete	

รูป 10 ซ้าย เลือก Gauge, กลาง เลือก PIN V1 , ขวา ค่า 0-100 และ /PIN.##/

# ติดตั้งไลบรารี Blynk บน Arduino IDE

เพื่อการใช้งาน Blynk จะต้องติดตั้งไลบรารีของ Blynk ก่อน ซึ่งให้ไปที่รายการ Sketch> Include Library > Manage Libraries... จะได้หน้าต่าง Library Manager ในหน้าต่างนี้ให้ค้น Blynk และเลือกติดตั้ง Blynk (คลิก Install)

All	v Top	pic All	~	Blynk	
lynk					
y Volodym	yr Shyman	skyy	iact in min	utes! It supports Willing I.F. Bluetooth, Ethernet, GCM, USB, Serial, Works with	
nany board	s like ESP8	266, ESP32, A	rduino UNC	D, Nano, Due, Mega, Zero, MKR100, Yun, Raspberry Pi, Particle, Energia, ARM	
	Edison/Gali	leo/Joule, BB	C micro:bit	, DFRobot, RedBearLab, Microduino, LinkIt ONE	

**รูป 11** หน้าต่าง Library Manager เลือก Blynk

ต่อไปก็ทดสอบ การทำงานของ Blynk การทดสอบนี้ยังไม่มีเซนเซอร์อะไรมาเกี่ยวข้อง เพียงทดสอบว่า ต่อเชื่อมกับ Blynk ได้หรือไม่

ให้ใส่ค่า auth[] ตามค่า รหัส ที่ Blynk ส่งมาให้ทางอีเมล์ ใส่ ssdi และ pass ตามที่กำหนด ในที่นี้ใช้โทรศัพท์มือถือ ปล่อยสัญญาน

**Code 3.** ทดสอบกับ ESP32/M5Stack

```
#define BLYNK_PRINT Serial
```

```
#include <WiFi.h>
#include <WiFiClient.h>
#include <BlynkSimpleEsp32.h>
// You should get Auth Token in the Blynk App.
char auth[] = "xIf2_yiKTAdAvqJcCyMq3rHBddxxxxx";
// Your WiFi credentials.
// Set password to "" for open networks.
char ssid[] = "OPPO A3s";
char pass[] = "123456789";
void setup(){
  Serial.begin(9600);
  Blynk.begin(auth, ssid, pass);
 delay(1000);
 Serial.println("Setting up.");
}
void loop(){
  Blynk.run();
  delay(9000);
 Serial.println("Blynk is running.");
}
```

เมื่อทดสอบโปรแกรมนี้ ให้เปิด Tools > Serial Monitor รอไว้ได้เลย ให้สังเกตว่า ใช้ ความเร็วอยู่ที่ 9600 baud ตามที่กำหนดไว้โปรแกรม Serial.begin(9600)

💿 СОМ7			_		×	
1					Send	
[1128]					1	~
/_)///_						
					_	
// v0.6.1 on ESP32						
[1202] Connecting to blynk-cloud.com:80						
[1433] Ready (ping: 84ms).						
Setting up.						
Blynk is running.						
Blynk is running.						
Blynk is running.						
						ø
Autoscroll Show timestamp	Newline	✓ 9600 ba	ud 🗸	Cle	ar outpu	t

**รูป 11** หน้าต่าง Serial Monitor แสดงการต่อเชื่อม Blynk

# ทดสอบใช้งานจริง

เมื่อทุกอย่างพร้อมแล้ว คือ 1) สร้าง แอปฯ Blynk ที่มีตัววัดค่า อุณหภูมิ และความชื้น 2) ต่อวงจร ESP32 กับ DTH22 3)ทดสอบการต่อ WiFi ได้แล้ว 4) ทดสอบการทำงานของ Blynk ก่อนหน้านี้เพื่อต่อเชื่อมตามรหัสที่ให้มากับอีเมล์ได้ แล้ว ต่อไปก็จะเป็นการเชื่อมต่อ WiFi ของ ESP32 กับ Blynk ตามตัวอย่างต่อไปนี้

```
#define BLYNK PRINT Serial
#include <WiFi.h>
#include <WiFiClient.h>
#include <BlynkSimpleEsp32.h>
#include "DHTesp.h"
DHTesp dht;
int dhtPin = 17;
// You should get Auth Token in the Blynk App.
char auth[] = "xIf2_yiKTAdAvqJcCyMq3rHBddxxxxx";
// Your WiFi credentials.
// Set password to "" for open networks.
char ssid[] = "OPPO A3s";
char pass[] = "123456789";
void setup()
{
  // Debug console
  Serial.begin(9600);
  Blynk.begin(auth, ssid, pass);
  delay(1000);
  Serial.println("Setting up.");
  dht.setup(dhtPin, DHTesp::DHT22);
}
float humi;
float temp;
void sendSensor(){
  Blynk.virtualWrite(V1, temp);
  Blynk.virtualWrite(V2, humi);
}
void loop()
{
  Blynk.run();
  delay(2000);
  Serial.println("Blynk is running.");
  TempAndHumidity newValues = dht.getTempAndHumidity();
  if(dht.getStatus()==0){
    TempAndHumidity newValues = dht.getTempAndHumidity();
  }
  Serial.println(" T:" + String(newValues.temperature) + " H:" +
String(newValues.humidity));
  temp = newValues.temperature;
  humi = newValues.humidity;
  sendSensor();
}
```

เมื่อรันโปรแกรมข้างต้นได้แล้ว ทดสอบการอ่านค่าผ่าน Serial Monitor เพื่ออ่านข้อมูลได้แล้ว ให้ทดสอบการอ่าน ผ่าน แอปฯ Blynk โดยกด เครื่องหมายมุมขวาสุด เพื่อเริ่มทำงาน และถ้าต้องการหยุด ก็กด เครื่องหมายขวาสุด ดังจะได้ผลดัง ร**ูป 12** 



รูป 12 ผลการทำงานของค่าวัด อุณหภูมิ และความชื้น