### 1. เริ่มต้น NodeMCU

## เนื้อหาประกอบด้วย

- ทำความรู้จัก NodeMCU
- การใช้ Arduino IDE กับ NodeMCU
- ทดลองเขียนโปรแกรมอย่าง่าย

การเขียนโปรแกรมกับบอร์ด NodeMCU เป็นสิ่งที่ไม่เหมือนกับการเขียนโปรแกรมกับคอมพิวเตอร์ เราใช้เขียนการทำงาน กับบอร์ด นี้ เพียงเพื่อควบคุม การรับ-ส่งข้อมูลของเซ็นเซอร์ต่างๆ เพื่อประมวลผลในพฤติกรรมต่างๆ ในความจำกัดของ ทรัพยากร ที่มีมากกว่า คอมพิวเตอร์ทั่วไป แต่ข้อจำกัดนี้ ก็มีข้อดี คือ เล็ก ใช้พลังงานไฟฟ้าน้อย ราคาย่อยเยา แต่ข้อเสียคือ เรา ไม่อาจจะทำให้ NodeMCU ประมวลผลรูปภาพได้ เราทำได้แค่งานเล็กๆ น้อย ในการรับ-ส่งข้อมูลพื้นฐานเท่านั้น การทำงานที่ หนักกว่านั้น จำเป็นต่อเชื่อมกับอุปกรณ์อื่น ๆ แต่การประมวลผลในระดับนี้ ก็จำเป็นสำหรับงานพื้นฐาน เช่น การควบคุม การ ปิด-เปิด อุปกรณ์อิเล็กทรอนิกส์ ซึ่งนำไปใช้ทำงานได้หลากหลายในชีวิตประจำวัน ไม่ว่าจะเป็นการ ให้เปิดกล่องวงจรปิด เริ่ม ทำงาน เมื่อมีผู้บุกรุกเข้าพื้นที่ตรวจสอบ แล้วส่งข้อมูลไปยังอินเตอร์เน็ต การตรวจสอบความชื้น แสง เพื่อการเปิด-ปิดน้ำ ในการ ดูแลต้นไม้ และอื่นๆ ที่จะประยุกต์ใช้ได้

## รู้จัก NodeMCU

NodeMCU เป็นบอร์ด Open Source/Hardware มี เฟิร์มแวร์ (firmware) เป็นฮาร์ดแวร์ที่บรรจุชอฟต์แวร์ ชื่อ ESP8266 เป็นชิปสนับสนุนการทำงานการต่อเชื่อม WiFi อย่างสมบรูณ์ ปัจจุบันกลายเป็นบอร์ดคอนโทรลเลอร์ ได้รับความ นิยมกันมาก เนื่องตัวมีขนาดเล็ก ที่มาพร้อมกับ WiFi และกระแสการใช้งานแบบ IOT ทำให้บอร์ด NodeMCU ตอบสนองความ ต้องการได้พอดี การต่อเชื่อมกับคอมพิวเตอร์ก็ใช้ สาย miniUSB ซึ่งเป็นแบบเดียวกับที่ใช้งานกับโทรศัพท์สมาร์ทโฟนทั่วไป

NodeMCU มีหลายรุ่น เช่น รุ่น 0.9 หรือเรียกว่า V1 กับ 1.0 หรือเรียกว่า V2 สองรุ่นมีขนาดที่ต่างกัน แต่ที่เหมาะ กับใช้งานกับ บอร์ดขนมปังมากกว่า คือรุ่น 1.0 เพราะมีพื้นที่เหลือให้ต่อขยายได้ ในขณะที่รุ่น 0.9 ใช้เต็มพื้นที่บอร์ดขนม ปัง นอกจากนี้มี อีกรุ่นหนึ่งคือ V3 มีขนาดใหญ่กว่า ซึ่งผลิตโดยบริษัท Lolin แต่ยังคงมีพิน เหมือนกับรุ่น 2



รูป 1. NodeMCU v0.9 กับ NodeMCU v1.0

ที่มา http://www.cnx-software.com/2015/10/29/getting-started-with-nodemcu-board-powered-by-esp8266-wisoc/

เอกสารประกอบการสอน วิทยาลัยเซาธ์อีสท์บางกอก หน้า 1

ตาราง 1 รุ่นของ NodeMcu

NodeMcu	V1	V2	V3
WiFi Module	ESP-12	ESP-12E	ESP-12E
USB to Serial	CH340	CP2102	CH340
ขนาด	เล็ก	กลาง	ใหญ่

#### อินพุท / เอาท์พุท

พินหรือขา ต่างๆ ของ NodeMCU ใช้เป็นแบบ GPIO (General Propose linput/output) เมื่อเทียบใช้งานกับ บอร์ด Arduino UNO จะมีขาที่ไม่ตรงกัน แต่ใช้เทียบกันได้ เช่น Pin 13 ของ Arduino จะมีค่าเท่ากับ ขา D7 ของ NodeMCU, ขา 12 ของ Arduino จะมีค่าเท่ากับ ขา D6 ของ NodeMCU เป็นต้น เราดูการเทียบ ขา ต่างๆ จากรูปที่ 2

ระบบการใช้ไฟฟ้า การทำงานของขาต่างๆ ทำงานกับไฟฟ้าแรงดัน 3.3 โวลต์ แต่กับ ขา Vin ที่รับไฟฟ้าแรงดัน 5 โวลต์ เช่นเดียวกับ ต่อกับสาย miniUSB ที่มีแรงดันไฟฟ้า 5 โวลต์

นอกจากนี้ยังมี **LED build-in** เทียบได้เป็น D0 หรือ GPIO16 สามารถนำมาทำทดสอบ LED โดยไม่ต้องมี LED มาต่อเพิ่ม ก็ได้ และยังมี LED อีกตัวหนึ่งที่แสดงสถานการณ์อัพโหลด



รูป 2 ขาของ NodeMCU V3 แจงในรูป GPIO (ที่มา: www.TheEngineeringProjects.com)

Arduino IDE (Arduino IDE - Arduino Integrated Development Environment)<sup>1</sup>

การเขียนโปรแกรมกับ NodeMcu ใช้ตัวพัฒนาเป็น Arduino IDE ซึ่งเป็นโปรแกรมพัฒนาที่ใช้กับบอร์ด อาร์ดูโน่ แต่ เนื่องด้วยเป็นโปรแกรมพัฒนาที่ใช้งานง่าย นักพัฒนา NodeMcu จึงเลือกใช้ โปรแกรมพัฒนานี้ด้วย ภาษาที่ใช้เขียนเรียกว่า C – Arduino

<sup>1</sup> ดาวน์โหลโปรแกรม Arduino IDE ได้ที่ https://www.arduino.cc

โปรแกรม Arduino-IDE เรียกกันว่า Sketch การติดตั้งสำหรับ ระบบปฏิบัติการ Windows มีให้เลือกแบบไฟล์ Installer ซึ่งติดตั้งอัตโนมัติ ในระหว่างติดตั้งไฟล์นี้ จะมีการถามให้ลงไดเวอร์ต่างๆ หลายครั้งให้ตอบตกลง และ Zip ไฟล์ เรียก ทำงานผ่าน ไฟล์ arduino.exe อาจต้องติดตั้ง Driver เอง ซึ่งจะมีมาด้วยกับที่ ดาวน์โหลด (ดูที่ Folder Driver)

File Edit Sketch To	ols Help	
sketch_mar10	Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload	<u>م</u>
void setup() // put your	Serial Monitor Ctrl+Shift+M Board: "Arduino Uno"	
}	Port	Serial ports
void loop() { // put your	Programmer: "AVRISP mkll"	COM3 COM4

รูป 3. Arduino IDE

# ติดตั้ง ไลบารี่ NodeMCU กับ Arduino IDE

การใช้งานกับ NodeMCU ได้จะต้องติดตั้งไลบรารี เพื่อทำงานได้กับ Arduino IDE ให้เลือกไปที่ File> Preferences ภายในหน้าต่าง Preferences นี้ให้เลือก Additional Boards Manager URLs: ซึ่งล่างสุดของหน้าต่างนี้ ให้ใส่ URLว่า "http://arduino.esp8266.com/stable/package\_esp8266com\_index.json" จากนั้นกด OK

	Preferences	
Settings Network		
Sketchbook location:		
C:\Users\pone\Documents\Ar	duino	Browse
Editor language:	English (English) v (requires restart of Arduino)	
Editor font size:	12	
Interface scale:	Automatic 100 0 % (requires restart of Arduino)	
Show verbose output during:	compilation upload	
Compiler warnings:	None 🗸	
Display line numbers		
Enable Code Folding		
Verify code after upload		
Use external editor	п	
Check for updates on star	tup	
Update sketch files to new	extension on save (.pde -> .ino)	
Save when verifying or up	loading	
Additional Boards Manager URI	Ls: http://arduino.esp8266.com/stable/package_esp8266com_index.json	C
Urra reafarantes can ha arbia	ส่งสามากรรม สาย	

เมื่อติดตั้งแรก Arduino IDE จะต้องดาวน์โหลด ไดรเวอร์บอร์ด จากไลบรารีอ้างอิง ที่สร้างก่อนหน้านี้ โดยการเลือก Board Manager แล้วให้เลื่อนหา esp8266 เพื่อทำการติดตั้ง (คลิก More Info และเลือก รุ่นล่าสุด แล้วติดตั้งโปรแกรม) เมื่อติดตั้งแล้ว จะขึ้นข้อความว่า INSTALLED ดังรูปต่อไปนี้

20		Boards Manager
Type All	v	
EMoRo 2560, Bo Online help More info	ard based on ATmega	₽ 2560 MCU.
AMEL-Tech Boar Boards included SmartEverything <u>Online help</u> <u>More Info</u>	ds by replaced by Arr in this package: Fox.	row Boards
esp8266 by ESP Boards included Generic ESP8266 Adafruit HUZZAH ESP-210, WeMos <u>Online help</u> <u>More info</u>	8266 Community vers in this package: : Module, Olimex MOD ESP8266 (ESP-12), Et : D1, WeMos D1 mini,	sion 2.3.0 INSTALLED >-WIFI-ESP8266(-DEV), NodeMCU 0.9 (ESP-12 Module), NodeMCU 1.0 (ESP-12E Module), SPresso Lite 1.0, ESPresso Lite 2.0, Phoenix 1.0, Phoenix 2.0, SparkFun Thing, SweetPer ESPino (ESP-12 Module), ESPino (WROOM-02 Module), WifInfo, ESPDuino.

รูป 5 esp8266 Library ที่ติดติดตั้งแล้ว

แต่เมื่อดาวน์โหลดมาพร้อมแล้ว จะพบบอร์ดให้เลือก ตามรูปต่อไปนี้ เป็น NodeMCU 1.0 (ESP 12E Module) และเลือก Port ในที่นี้เลือก Port 9 แต่ละเครื่องจะพบ Port ที่ต่างกันสามารถตรวจสอบได้ที่ Control Panel > System > Device Manager กรณีที่ไม่เห็น Com Port ที่ใช้กับ NodeMCU ให้ไปดาวน์โหลดมาติดตั้งเองได้ที่

https://www.silabs.com/products/development-tools/software/usb-to-uart-bridge-vcp-drivers

<u>©</u>	nodemcu_code   Arduino 1.8.	2 Hourly Build 2017	/03/08 02:33 🛛 🗕 🗖 📉
File Edit Sketch To	ools Help		
nodemcu_cod	Auto Format Archive Sketch Fix Encoding & Reload	Ctrl+T	© ▼
<pre>int led_d7 = void setup() pinMode(led</pre>	Serial Monitor Serial Plotter	Ctrl+Shift+M Ctrl+Shift+L	^
1	WiFi101 Firmware Updater		Δ
	Board: "NodeMCU 1.0 (ESP-12E Mo	dule)"	Boards Manager
<pre>void loop() {     digitalWrit     delay(1000)     digitalWrit</pre>	CPU Frequency: "80 MHz" Flash Size: "4M (3M SPIFFS)"		Arduino AVR Boards Arduino Yún Arduino/Genuino Uno

รูป 6 เลือก Board: "NodeMCU 1.0(ESP-12E Module)"



เมื่อติดตั้งแล้ว ทดลองสำรวจตัวอย่างโปรแกรมมากมาย ให้ไปที่ Arduino IDE แล้วเลือก File > Examples > ESP8266 ซึ่งสามารถทดลองใช้งานได้

New Open Open Recent	Ctrl+N Ctrl+O t ▶		2 5	
Sketchbook	•			^
Examples	9	. · · · · · · · · · · · · · · · · · · ·		
Close	Ctrl+W	Firmata		
Save	Ctrl+S	LiquidCrystal		
Save As	Ctrl+Shift+S	SD		
		Stepper		
Page Setup	Ctrl+Shift+P	Temboo	- 363	
Print	Ctrl+P	TFT	3.	
Preferences	Ctrl+Comma	WiFi	•	
Quit	Ctrl+Q	Examples for NodeMCU 1.0 (ESP-12E Module	Ň	
Concellator		ArduinoOTA		
		DNSServer		
		EEPROM		
		ESP8266	4	Blink
		ESP8266AVRISP		BlinkWithoutD
one uploading		ESP8266HTTPClient		CallSDKFunctio

รูป 8 ตัวอย่างโปรแกรมของ ESP8266

## เริ่มต้น Hello, World

เพื่อเป็นกำลังใจในการใช้งานครั้งแรก ก็เหมือนกับการเขียนโปรแกรมภาษาคอมพิวเตอร์อื่นๆ ที่มักเริ่มต้นเขียน โปรแกรมแรก ที่เรียกว่า Hello world ให้ เขียนโปรแกรมทักทายกัน เป็นการเขียนผ่าน Arduino IDE แล้วลงโปรแกรมที่ เขียนนี้ไปยังบอร์ด NodeMcu โดยยังไม่ต้องต่ออุปกรณ์อิเล็กทรอนิกส์อื่น ด้วยการทำตามลำดับการทดลองต่อไปนี้

- 1. ต่อเชื่อม USB ของ อาร์ดูโน กับ Computer
- เปิดโปรแกรม Arduino IDE ตรวจสอบ การต่อเชื่อม อาร์ดูโน กับ อาร์ดูโน IDE
   เช่น Tool> Port "9" ซึ่งแต่ละเครื่องได้ Port ไม่ตรงกัน ให้ตรวจสอบที่ Device Manager ใน Control panel >
   System ให้ดูที่ Ports(Com & LPT) จะพบ คำว่า Silicon Labs CP210x USB to UART Bridge (COM9)

💿 nodemcu_hell	o_world   Arduino 1.8.2 Hourly Build	2017/03/	□ ×
File Edit Sketch To	ools Help		
nodemcu_hell	Auto Format Archive Sketch Fix Encoding & Reload	Ctrl+T	₽. ■
void setup() // Initiate Serial.begi	Serial Monitor Serial Plotter WiFi101 Firmware Updater	Ctrl+Shift+M Ctrl+Shift+L	^
<pre>} void loop() [ Serial.prin }</pre>	Board: "NodeMCU 1.0 (ESP-12E Module)" CPU Frequency: "80 MHz" Flash Size: "4M (3M SPIFFS)" Upload Speed: "115200"	*	
	Port: "COM9"		Serial port
	Get Board Info		COM3
	Programmer: "AVRISP mkll" Burn Bootloader		COM4 COM9

3. เขียนโปรแกรม ดังนี้

```
void setup() {
    // Initiate serial communication
    Serial.begin(9600);//baud rate :9600 bit per second
}
void loop() {
    Serial.println("Hello");
}
```

4. กดปุ่ม Upload ให้ส่วนโปรแกรมนี้ ไปทำงานที่ บอร์ด NodeMcu แล้วรอผล จะมีสถานะว่า Done uploading



ตรวจสอบ การทำงาน

Tools > Serial Monitor

ซึ่งจะได้ผลคือ

<b>®</b>	COM9	-		×
			Se	nd
Hello				~
Hello				
Hel				$\sim$
✓ Autoscroll	No line ending 🔍	11	5200 b	aud

#### โครงสร้างโปรแกรม

จากที่ได้ทดลองเขียนโปรแกรม ของปฏิบัติการ 1 จะเห็นว่า โครงสร้างโปรแกรม จะมีสองฟังก์ชัน คือ

1. setup(){

}

ในส่วนฟังก์ชันนี้ จะมีพื้นที่ให้เขียนโปรแกรม อยู่คู่ของปีกการ ฟังก์ชันนี้จะเป็น กำหนดค่าเริ่มต้น ของโปรแกรมที่จะ ใช้ทำงานทั้งหมดของระบบ Arduino เช่น กำหนด Serial.begin(9600) เป็นกำหนดว่า ให้มีส่งข้อมูลในระดับ 9600 บิท ต่อวินาที่ การทำงานส่วนนี้จะทำงานเพียงครั้งเดียว ตอนเริ่มต้นโปรแกรมเท่านั้น

2. looo(

)

{

ในส่วนนี้จะโปรแกรมจะทำงาน วนซ้ำไม่รู้จบ จนกว่า ปิดโปรแกรม การทำงานของอุปกรณ์อิเล็กทรอนิกส์ เราจะให้ ทำงานตลอดเวลาอยู่แล้ว ในการทดลองที่ผ่านมา ให้พิมพ์ค่า Hello ตลอดเวลา

กรณีที่ต้องการ เขียนคำอธิบาย เพื่อความเข้าใจในภาษาที่เราเข้าใจได้ง่าย โดยไม่มีส่วนเกี่ยวข้องกับการทำงานของ โปรแกรม เราสามารถใช้เครื่อง "//" นำหน้าได้ ทำให้ทั้งบรรทัดนั้นถูกข้ามไป

#### บอร์ดขนมปัง

ก่อนที่ต่อวงจรอิเล็คทรอนิกส์ เราต้องรู้จัก บอร์ดขนมปัง หรือ เรียกว่าอย่างเป็นทางการว่า Solderless breadboard แต่บางครั้งอาจได้ยินบางคนเรียกว่า Solderless หรือบางคนก็เรียกว่า Breadboard ตัวบอร์ดนี้เราจะใช้เชื่อวงจรไว้ด้วยกัน แนวเส้นสีเหลือง (ดูรูปที่ 9) จะเป็นจุดที่ใช้เชื่อมต่อกัน สังเกตว่า แนวนอน เป็นแนวยาว จะเชื่อมต่อกันหมด หมายความว่า ถ้า เราใช้สายไฟ ต่อในจุดใดก็ได้ของแนวนอน จะถือว่าเชื่อมต่อกันอยู่ เช่น เดียวกับแนวตั้ง จะต่อจุดใดของแนวตั้งก็ถือว่าเชื่อมต่อ กันหมด ดังเราไม่ควรอย่างยิ่งที่จะต่อผิด เพราะจะทำให้บอร์ด NodeMCU เสียหายได้

		ě			•	•	-			•	•			•	•			•		•	•	•		1	1		•	•	•		1		•	•	•	•		1				•	•		-		•	÷	•	•		-		•	•	•	•		-						•	•	•	•				
	1	•	_	1	uni.	-	-	-	-	-	10			-	-			15	-	-	-	-	8		-	-	-	-	10	-	-	-	-	-	8		-	-	-		8	-	-	-	-		8	-	-	_	_	1	Q.	-	-	_	-	8		_	_	-	8	_	-	-	-	Þ				
- <b>4</b>	٠			1.4	٠	٠	4	1		٠	٠	4	۰.	٠		1	٠	٠	6.5	٠	٠	٠			1.4	۰.	٠	٠	٠	٠		٢.	•	٠	٠	٠		۲.1	• •	۰.	٠	٠	٠	٠	1	۰.	٠	٠	٠			1.4	۲.	٠	٠	٠	٠	٠			1.4	1.1	 • •	•	٠	٠	٠	٠				ł
$\sim 10$	٠			1.4	٠	÷		. 4		٠				٠		1.4	٠	÷	6.5	٠	÷	*			1.1	۰.	٠	٠	٠	٠		۲.	٠	٠	٠	٠			• •	•	•	٠	٠	٠	1	۰.	٠	٠	٠	٠		14	٢.	٠	٠	٠	٠	٠			1.4	1		• 1	٠	٠	٠	٠			14	ł
$\sim -$																									14	۰.	٠	٠	٠			۰.	٠	٠	٠	٠		6.5				٠	٠	٠	1	۰.	٠	٠	٠			14	۰.	٠	٠	÷	٠	٠		14	1.4	1.1		• •	٠	٠	٠	÷	14		14	ļ
																									1.4			٠	÷		. 4			٠	÷							٠	÷		. 4		٠	÷	÷			1.4			٠	÷	÷				1.4	1.4			٠	÷	٠			1.4	14	Į
		•			٠	÷				٠				•			•	*			÷	÷				• •	•	٠	٠	٠			•	٠	٠	٠			• •	•	•	٠	٠	٠		• •	٠	٠	٠	٠		• •		•	٠	٠	٠	٠					• •	•	٠	٠	٠	٠				ł

รูป 9. แนวการต่อเชื่อม ของบอร์ดขนมปัง (Breadboard)

เอกสารประกอบการสอน วิทยาลัยเซาธ์อีสท์บางกอก หน้า 7

ข้อระวัง การต่อวงจรผิด เช่น ต่ออุปกรณ์อิเล็คทรอนิคส์ สายไฟ(สายที่มีไฟเข้า) โดยไม่มีการต่อลง กราวด์ จะทำให้บอร์ดพังได้

# สร้างหลอดไฟ LED กระพริบ

มาถึงตอนนี้ก็พร้อมแล้ว ที่จะทำลองโปรแกรมเริ่มต้นอย่างง่าย โดยใช้ NodeMCU ใช้เป็นอุปกรณ์เปิด-ปิด LED ของ NodeMCU ใช้ D7 (GPIO13) แต่ ขาทั่วไปของ Arduino ใช้ขาที่ 13



รูป 10 NodeMCU ต่อกับ LED

```
Code 1.
const int LED_D7 = 13;//Or D7
void setup() {
    pinMode(LED_D7, OUTPUT);
}
void loop() {
    digitalWrite(LED_D7, HIGH);
    delay(1000);
    digitalWrite(LED_D7,LOW);
    delay(1000);
```

}

จากตัวอย่างโปรแกรมนี้ จะเห็นว่า โครงสร้างโปรแกรม ยังคงเหมือนเดิม แต่มีส่วนกำหนดค่าคงที่ ซึ่งใช้คีย์เวิร์ดว่า const เรามักใช้คีย์เวิร์ดนี้กำหนดค่าที่ไม่ต้องการเปลี่ยนแปลงใดๆ ตลอดการทำงานของโปรแกรมที่ทำงาน ชื่อตัวแปร LED\_D7 สำหรับค่าคงที่เป็นธรรมเนียมปฏิบัติจะใช้ อักษรตัวใหญ่หมด ซึ่งก็เหมือนกับภาษาเขียนโปรแกรมคอมพิวเตอร์อื่นๆ ภาษาที่ เขียนนี้เป็นภาษา C อาร์ดูโน ซึ่งตัวอักษรเล็กใหญ่ไม่เหมือนกัน เช่น a ไม่เป็นตัวเดียวกับ A

LED ขายาวเป็น + ขาสั้นเป็น – หรือกราวด์

ในตัวอย่างนี้กำหนดให้ค่าคงที่เท่ากับ 13 (เลขจำนวนเต็ม ตรงกับชนิดข้อมูล int) ซึ่งตรงกับ ขาดิจิตอลที่ D7 และตัว แปรค่าคงที่นี้จะนำไปใช้ได้กับฟังก์ชัน setup( ) และ loop( )

การกำหนดค่าเริ่มต้น เพื่อระบุว่า LED ขาที่ 13 เป็นผลลัพธ์ หรือที่ให้ไฟเข้า ให้สังเกตว่าจะเป็นขายาว จะถูกกำหนด ที่ ฟังก์ชัน setup( ) และขาสั้นต่อกับ GND

สำหรับการทำงานซ้ำในฟังก์ชัน loop( ) เป็นทำให้ไฟ LED เปิด-ปิด โดยคำสั่งสำหรับให้ LED ติดไฟ และดับไฟ ใช้ การวัดแรงดันไฟฟ้าสูง ซึ่งทำให้ไฟติด หรือต่ำ ซึ่งทำไฟดับ โดยเขียนด้วยคำสั่ง digitalWrite( ) มีตัวแปรเข้าเป็น HIGH กับ LOW และถ้าหน่วงเวลาไป 1 วินาทีก็ทำได้ด้วยกำหนด delay(1000) ดังตัวอย่างที่ผ่านมา การทำงานนี้จะวนซ้ำไปเลื่อยๆ

### กฎของโอห์ม

เพื่อทำความเข้าใจเล็กน้อย เกี่ยวการไฟฟ้า ที่ไหลในวงจรทำงานอย่างไรกัน เพราะจะส่วนสำคัญต่อการใช้งาน อุปกรณ์อิเล็กทรอนิกส์ อย่างน้อยก็เพื่อเลือกใช้งานไม่เกิดความเสียหายได้

ก่อนอื่นให้นึกถึง ระบบน้ำประปาที่บ้าน โดยเปรียบเทียบว่า กระแสน้ำ ที่ไหลตามท่อ ปริมาณน้ำที่ไหลนั้น เปรียบเสมือนกระแสไฟฟ้า ซึ่งมีหน่วยวัดเป็น แอมแปร์ (Amperes) หรือเรียกสั้นว่า แอมป์ เทียบแรงดันน้ำเป็นแรงดันไฟฟ้า มี หน่วยเป็น โวลต์ (Volt) และเทียบ ความต้านทาน เป็นเหมือนท่อน้ำ มีหน่วยเป็น โอห์ม (Ohm) ใช้สัญลักษณ์ Ω แทนหน่วยนี้ ได้ หลักการคำนวณ ค่าเหล่านี้ มีสูตรสั้นๆ ว่า

$$V = I \times R \tag{1}$$

โดยที่ V แทนโวลต์ I แทนกระแส และ R แทน ความต้านทาน เราจำเพียงสูตร (1) ก็พอ การหาค่า I หรือ R ก็มาจาก กลับไป-มาของสูตรนี้ ลองพิจารณาสูตรต่อไปนี้ ที่มาจากการแปลงจากสูตรแรก

I

จากสูตร (1) นี้ ถ้าเพิ่ม กระแส (I) หรือ ความต้านทาน (R) ก็จะทำให้ แรงดัน (V) สูงขึ้น เพราะเป็นมาจากผลคูณของ ตัวนี้ สำหรับสูตร (2) ถ้า ต้องการให้ กระแส มีมาก ก็ต้องให้ตัวหาร (R) มีค่าน้อยๆ หรือตัวถูกหาร (V) มีค่ามากๆ สำหรับสูตร (3) เราก็วิเคราะห์ในทำนองเดียวกัน

> การที่ไฟฟ้าลัดวงจร ที่อาจทำให้อุปกรณ์ทดลองเสียหายได้ เนื่องจาก R เป็น 0 ทำ I มีค่าสูงสุด หรือค่า Infinity จากสูตร I = V/0

หรือจะใช้รูปสามเหลี่ยมแทนกฎของโอห์มก็จะช่วยให้จำได้ง่ายขึ้น ตามรูปนี้ ถ้าใช้หาค่า V ก็ดูส่วนล่าง ได้ผลเป็น I x R ถ้าหาค่า I ก็ให้ดูรูปส่วนขวา จะได้ V/R



ดังลองให้เปรียบเทียบระบบประปา ถ้าต้องการให้น้ำไหลแรง เราก็ต้องมีเครื่องปั้มน้ำ เปรียบได้กับ ถ่ายไฟฉาย ที่เป็น ที่เก็บไฟฟ้า เพื่อน้ำไหลต่อวินาทีแรงขึ้น ซึ่งส่งผลต่อระบบอื่นๆ ด้วย ที่ต้องรองรับความแรงของน้ำได้ เช่น ถ้าข้อต่อ ถ้าต่อกันไม่ ดี น้ำอาจรั่วได้ เช่นเดียวกัน ในอุปกรณ์อิเล็กทรอนิกส์ อุปกรณ์ต่างๆ ที่รับแรงดันไฟสูงมากเกินพิกัดความสามรถ ถ้าไม่ได้ทำมา เพื่อรองรับแรงดันไฟสูงได้ อาจทำให้อุปกรณ์นั้นเสียหายได้ นอกจากนี้ ปริมาณน้ำ หรือกระแสน้ำ ที่มีปริมาณมาก อาจทำให้ อุปกรณ์ทำงานตลอดเวลา ทำให้อาจอายุการงานสั้นลงได้ เหมือนกับหลอดไฟ หรือ LED ที่ต่อตรงกับบอร์ด Arduino หรือ NodeMcu ให้ทำงานนานๆ อาจเสียหายได้ เราต้องลดกระแสไฟลง วิธีการคือต้องมี ตัวต้านทาน ในระบบประปา มีก็อกน้ำ ถ้า เปิดน้อย น้ำก็ไหลน้อย ถ้าเปิดมากน้ำก็ไหลมาก รวมท่อประปาก็ถือเป็นต้านทานอย่างหนึ่งได้ ดังนั้นแล้วจึงเปรียบเหมือนตัว ต้านทานให้น้ำไหลได้มากหรือน้อย ในวงจรไฟฟ้า มีอุปกรณ์ที่ทำงานในลักษณะนี้ คือ ตัวต้านทาน (Resistor) ดังเพื่อลด กระแสไฟฟ้า ที่ทำงานบน LED ไม่ให้ทำงานที่ปริมาณไฟมากเกินไป เหมือนหลอดไฟ ที่รับปริมาณไฟมากๆ อาจร้อนไว และอายุ การใช้งานก็สั้นลง เราจึงต้องเพิ่มตัวต้านทานเข้าไปในวงจรเพื่อลดกระแสไฟฟ้า

เช่น การทดลองต่อ LED ที่ผ่านมา ใช้ การต่อ โดยไม่มีตัวต้านใดคั้น ทำให้มีกระไฟฟ้าไหลผ่านเต็ม ที่ ดังสูตรจากกฎ ของโอห์ม V = IR ถ้าต้องการหาค่ากระแส จะกลับสูตรเป็น I = V/R ซึ่งแปลได้ว่ายิ่งมีค่า R สูง จะได้กระแสต่ำ และถ้าค่า R ต่ำ จะมีกระแสสูง จากสูตรนี้เราสามารถนำไปคำนวณหาค่า R ที่เหมาะสมที่ใช้กับ LED ได้

ในกรณี การหา R ที่เหมาะสม เมื่อของ NodeMcu ไฟฟ้ามีแรงดัน 3.3 V และ LED ทั่วไปใช้แรงดันฟ้าตกคล่อม ประมาณเกิน 2.3 V โดยใช้กระแสประมาณ 20 mA ตัวต้านทาน ควร มีขนาด 135 โอห์ม คำนวณได้จากสูตร R = ( V<sub>cc</sub> - V<sub>f</sub>)/ I<sub>f</sub> = (3.3-**2.3)/0.02 = 5**0 โอห์ม

สำหรับบอร์ด NodeMcu จะใช้ GRD แทน กราวด์ หรือแทนการต่อสายดิน หรือแทนขั้วลบ และ พิน (PIN) ที่มี แรงดันไฟ จะแทนขั้วบวก การไหลของกระแสไฟฟ้า ในวงจรอิเล็กทรอนิกส์ (แบบธรรมเนียมนิยม) จะไหลจากขั้วบวก ไปยังขั้ว ลบ

#### แบบฝึกหัด

- 1. ให้เขียนโปรแกรมให้ หลอด LED กระพริบ ทุกวินาที โดยใช้ LED-build-in (GPIO16)
- ให้เขียนโปรแกรมให้ หลอด LED สองตัว สลับกันติด-ดับ