

# บทที่ 4

## ESP32

## Node MCU

อาจารย์สกรณ์ บุชบง

สาขาวิทยาการคอมพิวเตอร์

คณะวิทยาศาสตร์

มหาวิทยาลัยราชภัฏบุรีรัมย์

zagon.bb@bru.ac.th

## What is NODEMCU



**Connect Things EASY**

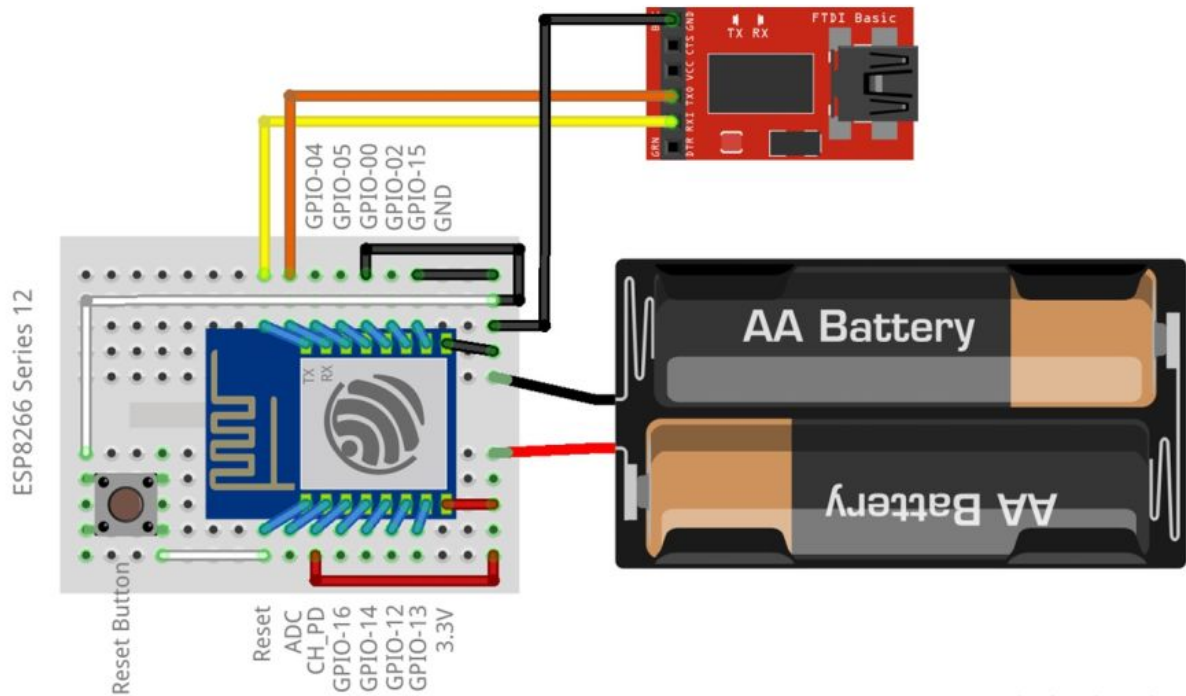
**“An open-source firmware and development kit that helps you to prototype your IOT product within a few Lua script lines.”**

## NodeMCU คืออะไร ?

NodeMCU คือ แพลตฟอร์มที่ใช้ในการพัฒนา Internet of Things (IoT) ตัวบอร์ดได้รวมโมดูล ESP8266, USBto-TTL และ พอร์ต micro USB เข้าด้วยกัน สามารถเขียนโปรแกรมด้วยภาษา Lua และ ภาษา C/C++ บน Arduino IDE ได้



# การเชื่อมต่อโมดูล ESP-12E กับ USB

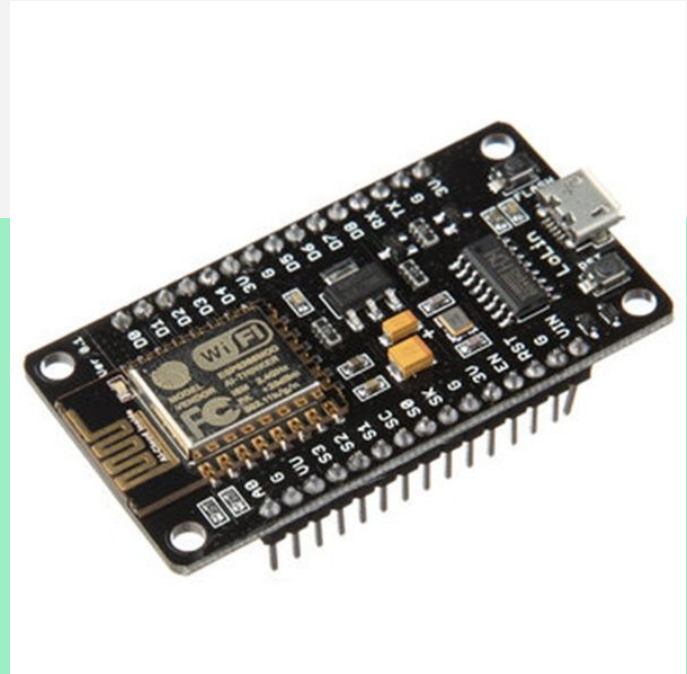


ESP8266 Series 12 Programming Circuit ( Gareth aka Chiprobot )



## NodeMCU คืออะไร ?

ตัวโมดูล ESP8266 นั้นมีอยู่ด้วยกันหลายรุ่นมาก ตั้งแต่เวอร์ชันแรกที่เป็น ESP-01 ไล่ไปเรื่อยๆจนปัจจุบันมีถึง ESP-12 แล้ว และที่ฝังอยู่ใน NodeMCU version แรกนั้นก็จะเป็น ESP-12 แต่ใน version 2 นั้นจะใช้เป็น ESP-12E แทน ซึ่งการใช้งานโดยรวมก็ไม่แตกต่างกันมากนัก



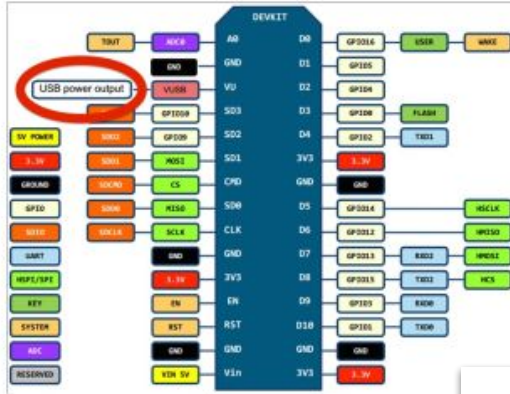
## NodeMCU คืออะไร ?

- NodeMCU นั้นมีลักษณะคล้ายกับ Arduino ตรงที่มีพอร์ต Input Output builtin มาในตัว
- สามารถเขียนโปรแกรมคอนโทรลอุปกรณ์ I/O ได้โดยไม่ต้องผ่านอุปกรณ์อื่นๆ
- มีนักพัฒนาที่สามารถทำให้ Arduino IDE ใช้งานร่วมกับ Node MCU ได้
  - จึงทำให้ใช้ภาษา C/C++ ในการเขียนโปรแกรมได้
  - ทำให้เราสามารถใช้งานมันได้หลากหลายมากยิ่งขึ้น
- NodeMCU ตัวนี้สามารถทำอะไรได้หลายอย่างมากโดยเฉพาะเรื่องที่เกี่ยวข้องกับ IoT ไม่ว่าจะเป็นการทำ Web Server ขนาดเล็ก การควบคุมการเปิดปิดไฟผ่าน WiFi และอื่น ๆ อีกมากมาย





# NodeMCU Version 3



## Compare NodeMCU size



## Nodemcu v3 module



ขยายขาให้เชื่อมต่อได้ง่ายขึ้น และรองรับการจ่ายไฟ  
6-24 VDC ได้



# แนะนำ ESP32

รู้จัก ESP32 และเปรียบเทียบกับ ESP8266

## แนะนำ ESP32

ESP32 เป็นชื่อของไอซีไมโครคอนโทรลเลอร์ที่รองรับการเชื่อมต่อ WiFi มีความสามารถการเชื่อมต่อ Bluetooth Low-Energy (BLE, BT4.0, Bluetooth Smart) ผลิตโดยบริษัท Espressif จากประเทศจีน



## ESP32 ได้แก้ไขจุดด้อยต่าง ๆ ของ esp8266 ไปจนหมด

- I/O และ Analog input ที่มีไม่เพียงพอกับการใช้งาน
- ปรับสเปคของ hardware ให้สูงขึ้น มีความเสถียรภาพสูง

- **Key Features**

- 240 MHz dual core Tensilica LX6 microcontroller with 600 DMIPS
- Integrated 520 KB SRAM
- Integrated 802.11 b/g/n HT40 Wi-Fi transceiver, baseband, stack and LwIP
- Integrated dual mode Bluetooth (classic and BLE)
- 16 MB flash, memory-mapped to the CPU code space
- 2.3V to 3.6V operating voltage
- -40°C to +125°C operating temperature
- On-board PCB antenna / IPEX connector for external antenna

- **Security Related**

- WEP, WPA/WPA2 PSK/Enterprise
- Hardware-accelerated encryption: AES/SHA2/Elliptical
- Curve Cryptography/RSA-4096

- **Performance**

- Max data rate of 150 Mbps@11n HT40, 72 Mbps@11n HT20, 54
- Maximum transmit power of 19.5 dBm@11b, 16.5 dBm@11g, 15.5 dBm@11n
- Minimum receiver sensitivity of -97 dBm
- 5  $\mu$ A power consumption in Deep-sleep

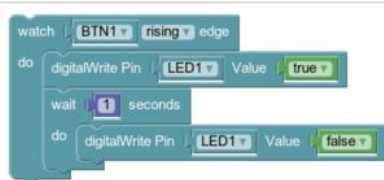
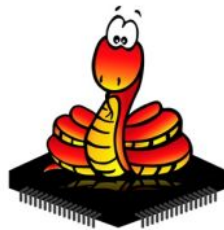
# ซอฟต์แวร์ที่ใช้พัฒนา

Arduino IDE

## Software Development



MicroPython



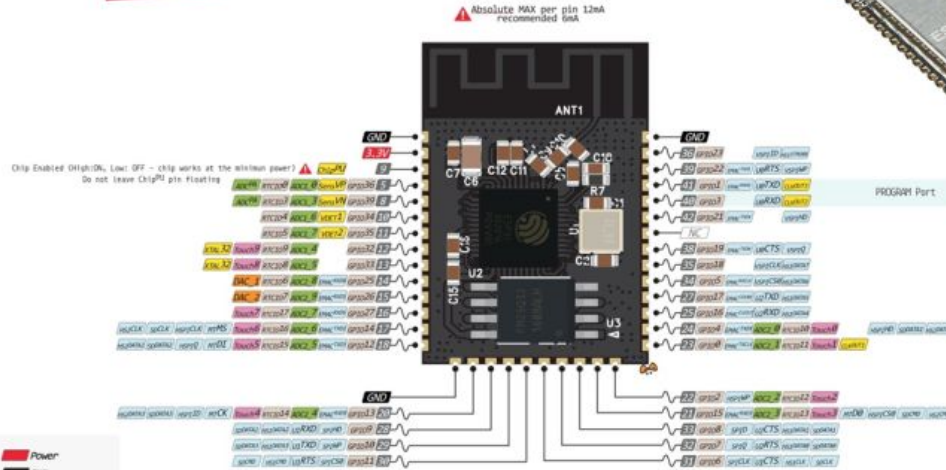
The Graphical Editor lets you use Esprimo even if you can't code!

The optional graphical editor makes it quick and easy to create simple devices, even if you've never programmed before.



# โมดูล ESP 32

## WROOM32 PINOUT

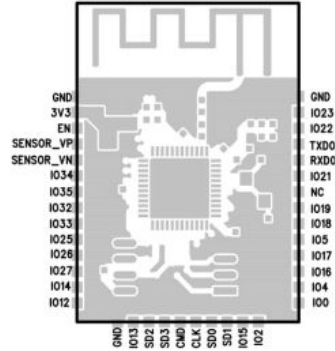


- Power
- GND
- Serial Pin
- Analog Pin
- Control
- Physical Pin
- Port Pin
- Touch Pin
- DAC Pin
- PWM Pin



04 Aug 2016  
ver 3 rev 0

## ESP32 Module – ESP-32S



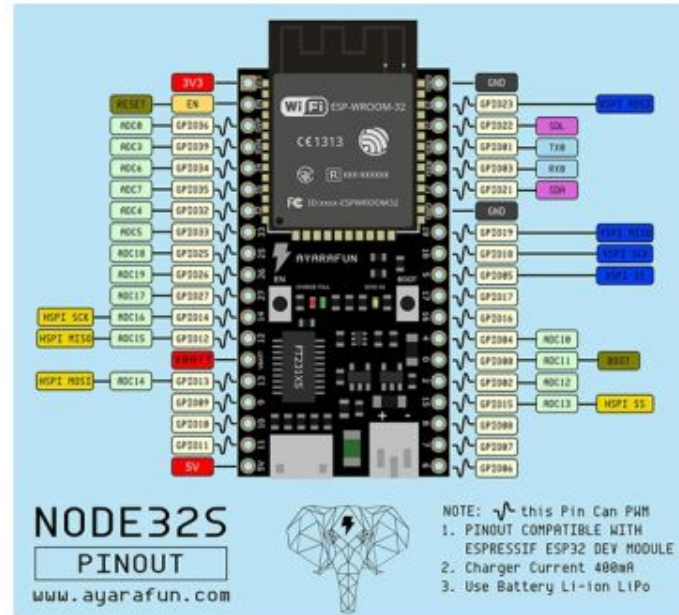
## EESP32 Module – ESP32-Bit



U1			
ESP32-Bit			
1	GND	VCC3V3	30
2	Sens_VP	GND	29
3	Sens_VN	GPIO21	28
4	CHIP_UP	TXD0	27
5	GPIO34	RXD0	26
6	GPIO35	GPIO22	25
7	GPIO32	GPIO19	24
8	GPIO33	GPIO23	23
9	GPIO25	GPIO18	22
10	GPIO26	GPIO5	21
11	GPIO27	GPIO17	20
12	GPIO14	GPIO16	19
13	GPIO12	GPIO4	18
14	GPIO13	GPIO0/DL	17
15	GPIO15	GPIO2	16

ESP32-Bit

# บอร์ดพัฒนา ESP32



## เปรียบเทียบระหว่าง ESP8266 และ ESP32

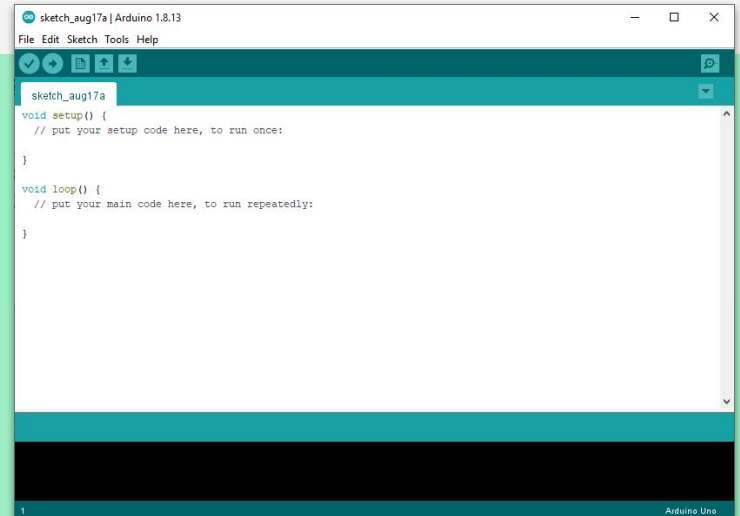
	ESP8266	ESP32
	Tensilica L106 32-bit single core micro controller	Xtensa® Single-/Dual-core 32-bit LX6 microprocessor(s)
ROM	none	448 KB
RAM	160kB	520 KB
RTC RAM	768 Bytes	8kB slow + 8kB fast
QSPI flash/SRAM,	up to 1 x 16 MB	up to 4 x 16 MB
GPIOs	17	36
Wi-Fi	802.11 b/g/n/e/i	802.11 b/g/n/e/i
Bluetooth	no	yes
PWM Pins	8 Software	1 Hardware/16 software
ADC	10 bit	12 bit
DAC	no	2 x 8 bit
Touch sensor	no	10
Temperature Sensor	no	yes
Cryptographic engine	no	yes
• Power supply:	2.5V to 3.6V	2.3V to 3.6V

# การติดตั้ง Arduino ide

เครื่องมือตัวหลักสำหรับการเขียนโปรแกรม

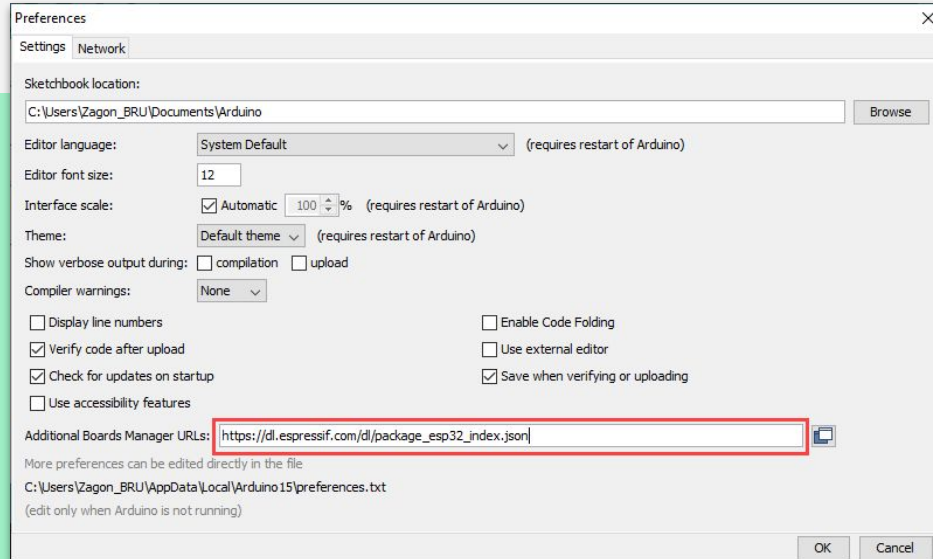
# การติดตั้ง Arduino IDE

1. <https://www.arduino.cc/en/main/software>
2. Download Windows Installer, for Windows 7 and up
3. install



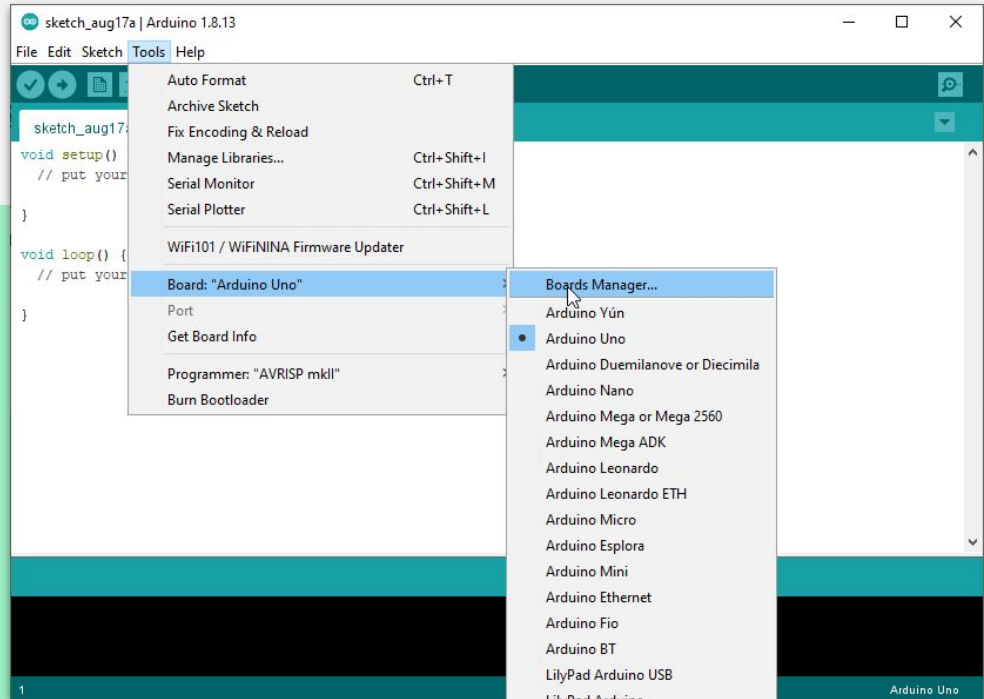
# การติดตั้ง ESP32 ใน Arduino IDE

- On arduino ide click **File -> Preferences**
- Add “[https://dl.espressif.com/dl/package\\_esp32\\_index.json](https://dl.espressif.com/dl/package_esp32_index.json)” to Additional Boards Manager URLs



# การติดตั้ง ESP32 ใน Arduino IDE

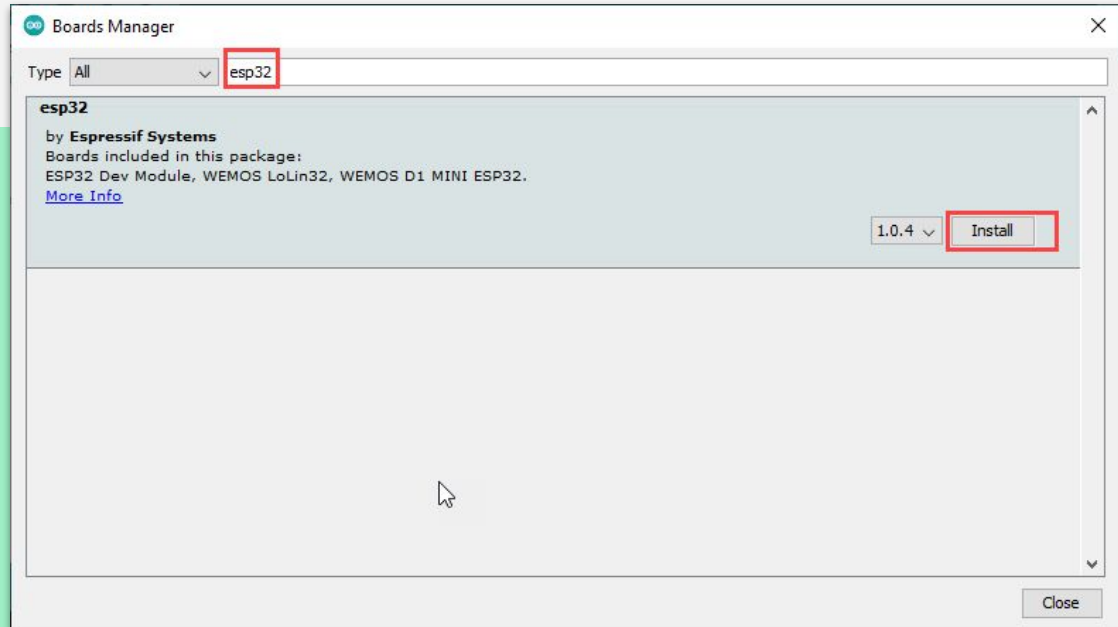
- Tools -> Board -> Board Manager





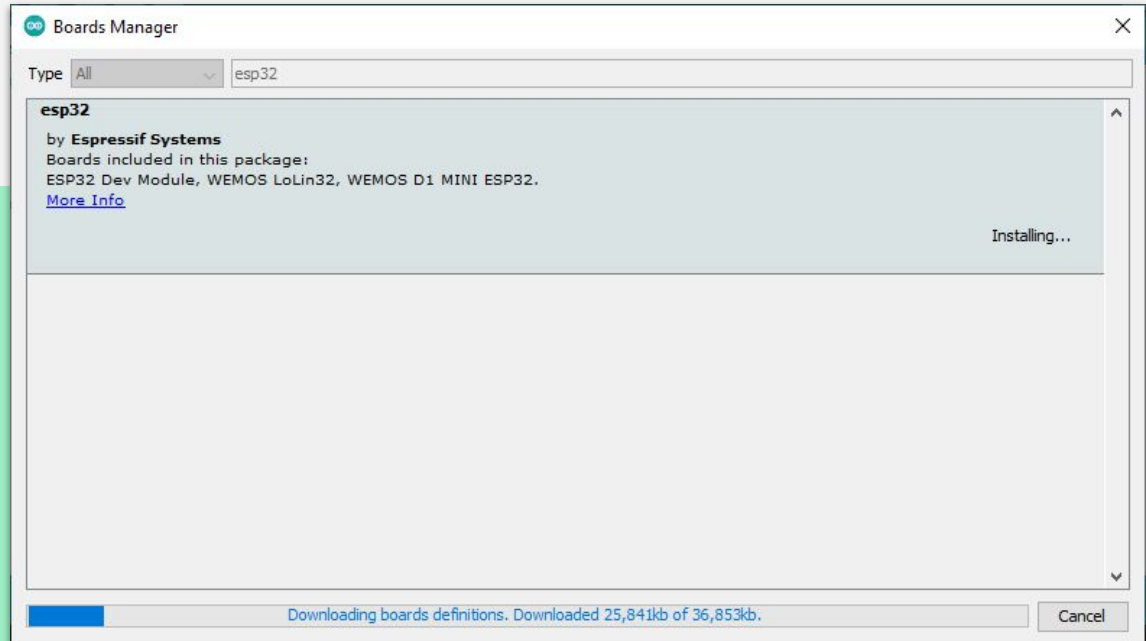
# การติดตั้ง ESP32 ใน Arduino IDE

- Typing “esp32” to search bar and click install



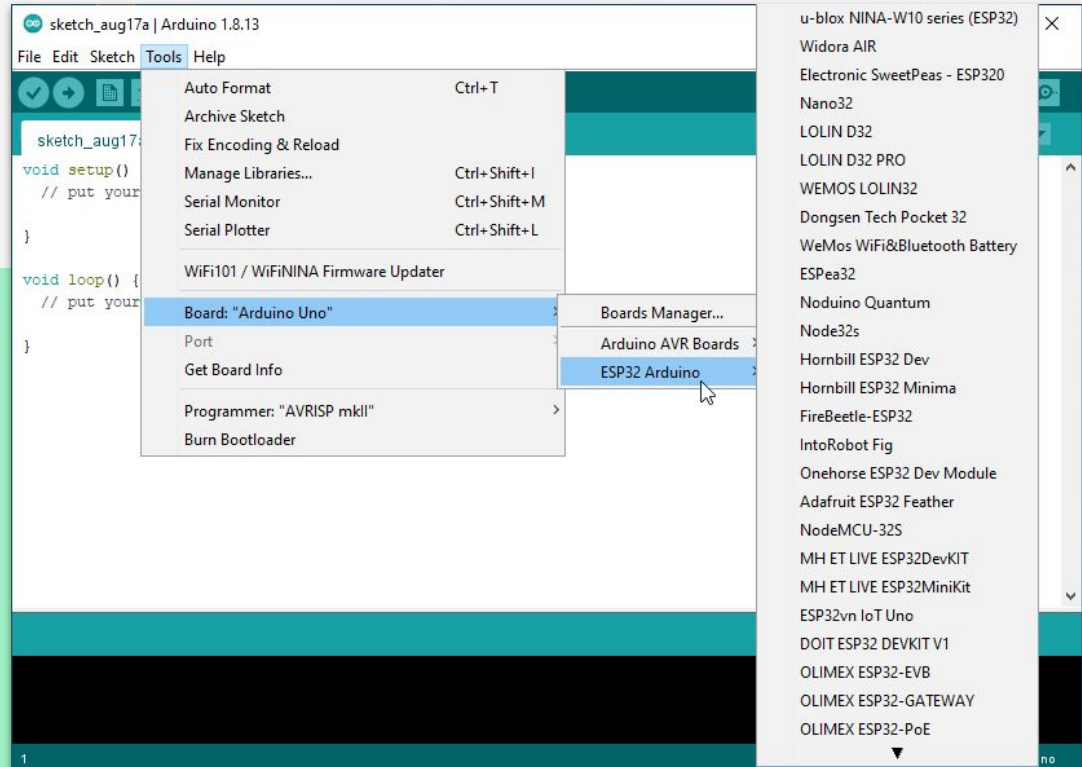
# การติดตั้ง ESP32 ใน Arduino IDE

- Wait until finished

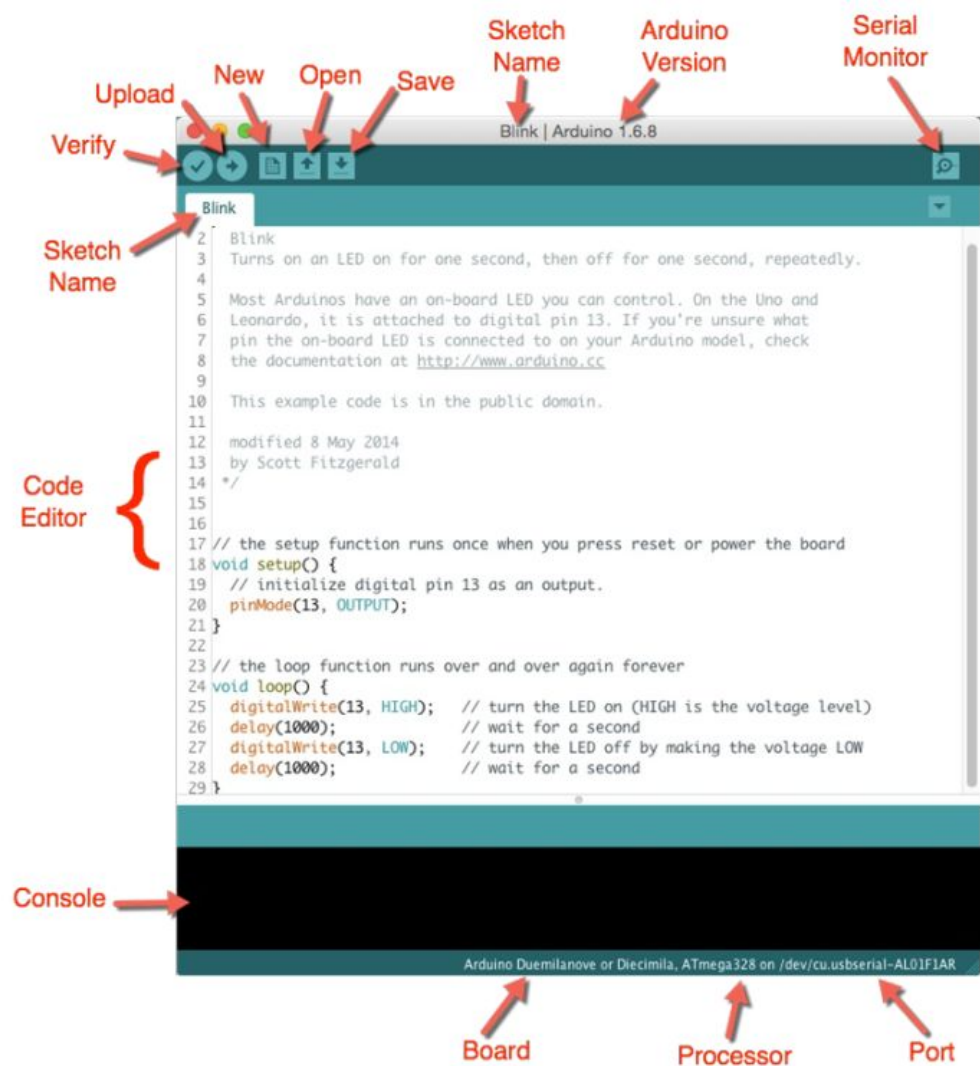


# การติดตั้ง ESP32 ใน Arduino IDE

- Checking Board ESP32 in Boards Manager



# Arduino IDE



## โครงสร้างภาษา C สำหรับ Arduino เบื้องต้น

```
sketch_jun21b | Arduino 1.6.9  
sketch_jun21b  
1 void setup() {  
2 // put your setup code here, to run once:  
3  
4 }  
5  
6 void loop() {  
7 // put your main code here, to run repeatedly:  
8  
9 }
```

\* ฟังก์ชัน setup() และ loop() จำเป็นต้องมีทุกโปรแกรม

9 Generic ESP8266 Module, 80 MHz, 40MHz, DIO, 115200, 512K (64K SPIFFS), ck, Disabled, None on /dev/cu.SLAB\_USBtoUART

Blink 5

```
1 /*
2  Blink
3  Turns on an LED on for one second,
4  then off for one second, repeatedly.
5  */
6
7 int Pin = 13;
8
9 // the setup function runs once when you press reset or power the board
10 void setup() {
11   // initialize digital pin 13 as an output.
12   pinMode(Pin, OUTPUT);
13 }
14
15 // the loop function runs over and over again forever
16 void loop() {
17   digitalWrite(Pin, HIGH); // turn the LED on (HIGH is the voltage level)
18   delay(1000);             // wait for a second
19   digitalWrite(Pin, LOW);  // turn the LED off by making the voltage LOW
20   delay(1000);             // wait for a second
21 }
```

# ฉบับที่ 4