

<https://www.facebook.com/lamloeicom>

คอร์สอบรม

Node32s Plus Netpie

Arduino IDE Basic

ขั้นพื้นฐาน



จุดประสงค์

- ผู้เข้าอบรมสามารถเขียน Arduino IDE ลงบน Node32s
- สามารถเชื่อมต่อ netpie ได้

เอกสารคอร์สอบรมนี้ สามารถดาวน์โหลดได้ที่

- <https://github.com/lamloei/present>

*** คอร์สอบรมนี้ หมายความว่า สมกับผู้เริ่มต้น ***



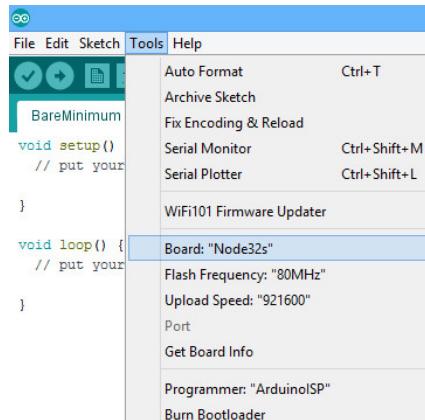
กำหนดการ

- 11 – intro & install
- 12 – basic input output
- 13 – lunch break
- 14 – adv IO touch
- 15 – sensor
- 16 – ble simple & connect wifi & temp & humi
- 17 – connect netpie



พื้นฐาน

Tools > Board: “Node32s”

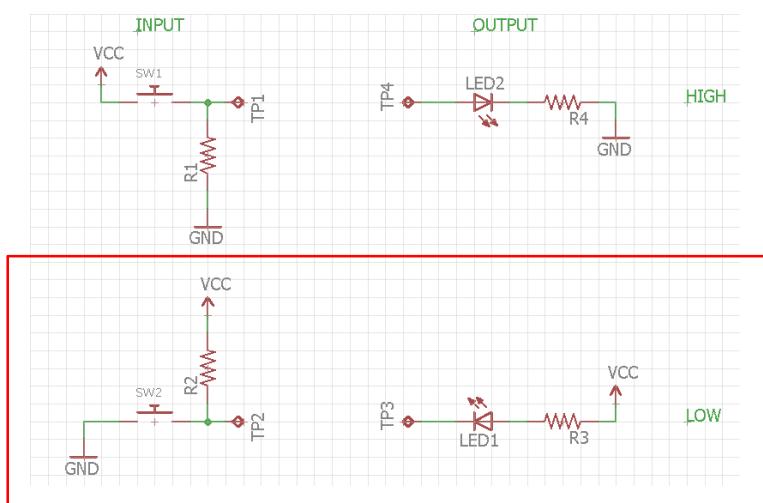


1. * ต้องกดไฟมด้า
2. เสียบสาย usb บอร์ด ต่อเข้าคอม
3. จะต้องเห็น COM port
4. ถ้ายังไม่ได้ติดตั้งปิร์แกรมให้ข้ามไปติดตั้ง
5. ที่ Arduino IDE ไปที่เมนู Tools
จะต้องเห็น Board กับ Port

1. สมัครสมาชิก Netpie.io



Active High Active Low



Active Low จะใช้แหล่งจ่ายภายนอก ทำให้มีเป็นภาวะของ MCU

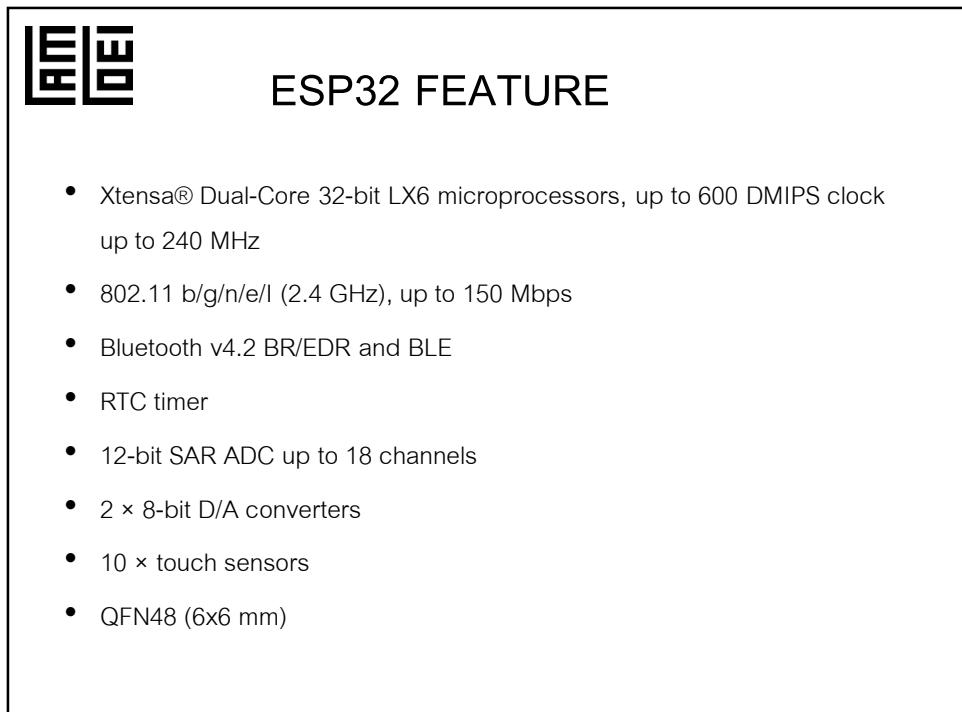
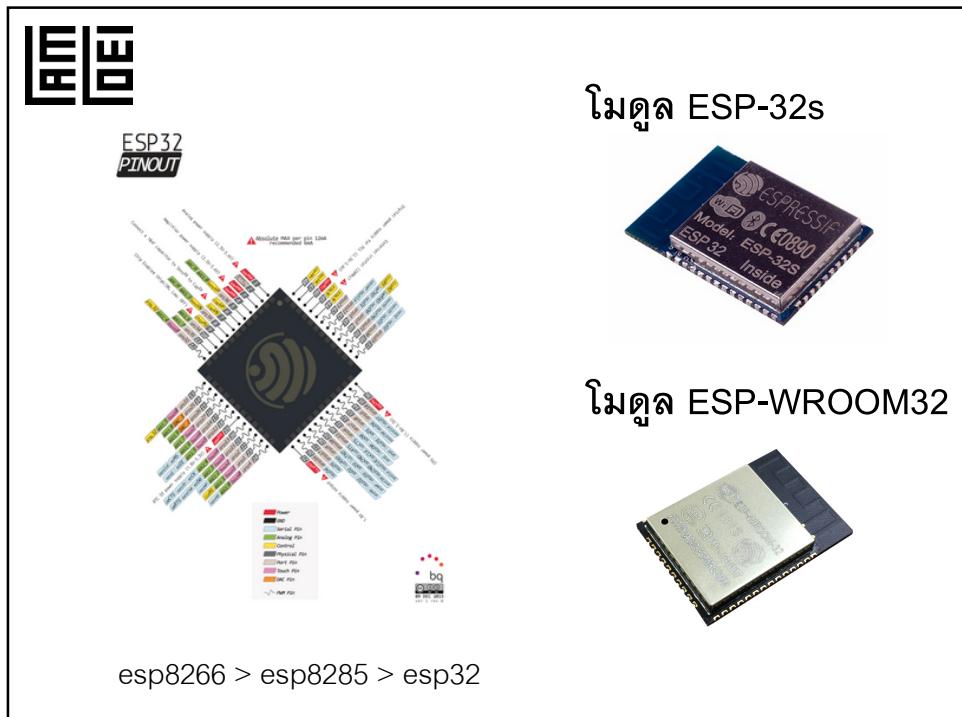
ແນະນຳຕ້ວ

- ຂຶ້ອ ຂຶ້ອເລັ່ນ
- ບຣິ່ຈັກ/ຫນ່ວຍງານ
- ຄວາມຄົນດີ ອີ່ອງການທີ່ທຳ
- ເຄຍໄສ້ esp8266, esp8285, esp32 ຕະກູດ ESP, NODEMCU ມາກ່ອນ
- ຄວາມຄາດຫວັງຂອງໂຄວັ້ງສອບວຸນນີ້

ESPRESSIF

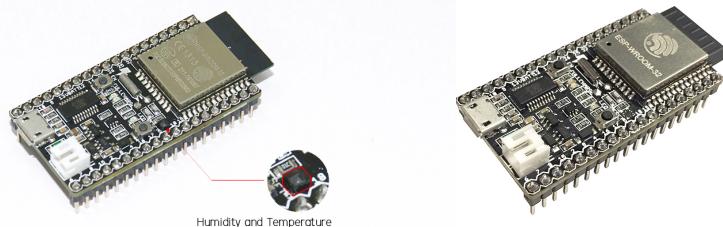
<https://espressif.com/>







Node32s Plus vs Node32s



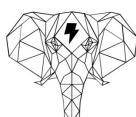
hts221

- ตัววัดค่าอุณหภูมิอยู่บนบอร์ด
- ทำให้ค่าที่ได้ร่วมอุณหภูมิบอร์ดร่วมไปด้วย มีผลให้ความซึ้งต่ำ และอุณหภูมิสูง กว่าปกติ



**NODE32S
PINOUT**

www/ayarafun.com



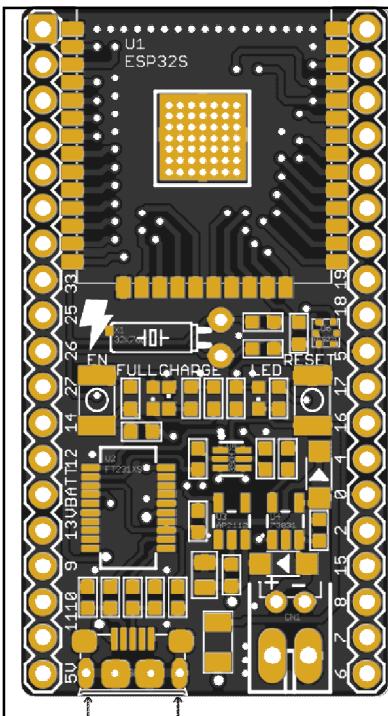
www.1amloe1.com

NOTE: ✓ this Pin Can PWM
 1. PINOUT COMPATIBLE WITH
 ESPRESSIF ESP32 DEV MODULE
 2. Charger Current 400mA
 3. Use Battery Li-ion LiPo

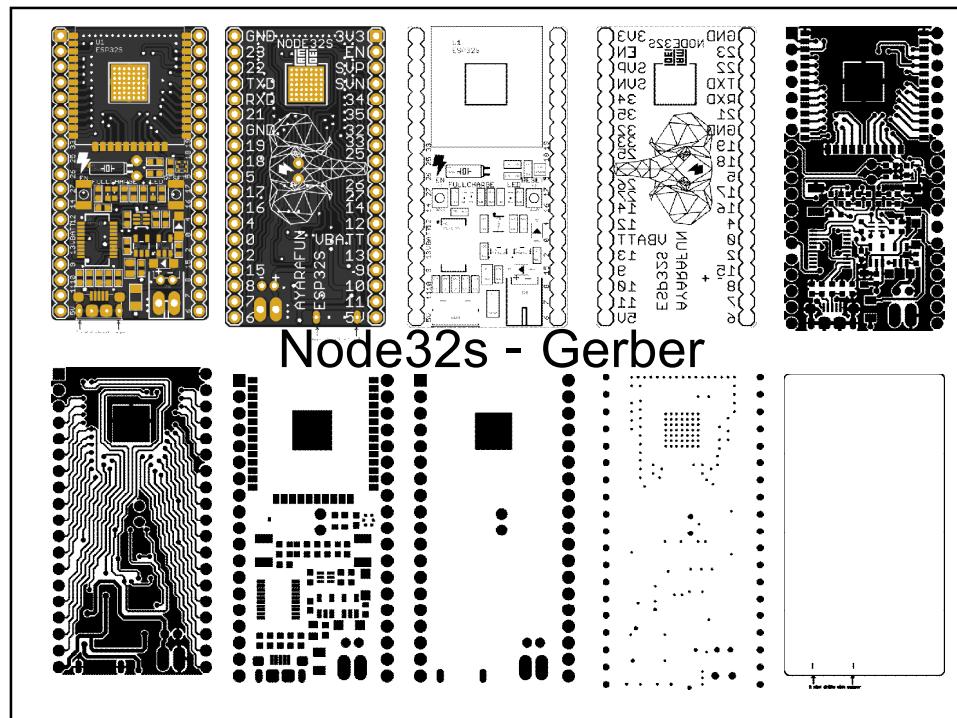
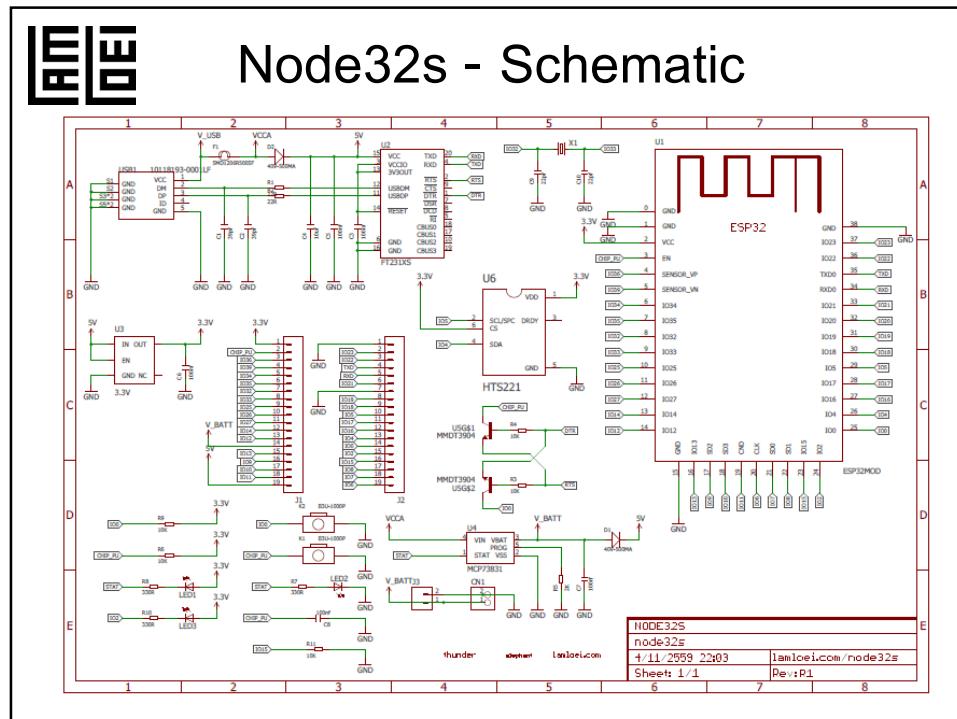


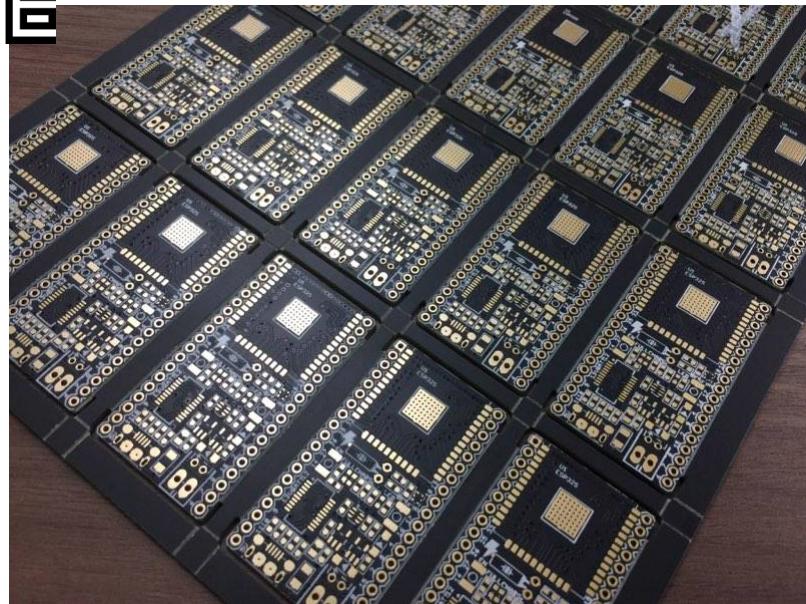
Reverse Pin

- Input only – 36, 39, 34, 35
- Not Used – 6, 7, 8, 9, 10, 11
- IO Special – 0, 2, 15



1. รองรับ esp-wroom-32 และ esp-32s
2. กว้างยาว 19 pin * 10 pin
3. JST 2mm 2pin connector
4. PTC Fuse 500mA
5. Micro usb FT231XS
6. Hts221 – temp & humidity
7. Ap2112 – Voltage regulator
8. Mcp73831 – charge battery
9. Mmdt3904 – dual transistor
10. Manual button – EN & io0
11. Crystal 32.768khz 26H





ESP-NESEMU prototype



พอร์จุนชั้น 4



จุดเด่น Node32s

- สามารถต่อ WiFi ได้
- ราคาถูก
- และใช้ Arduino IDE เขียนโปรแกรมได้



วิธีติดตั้ง

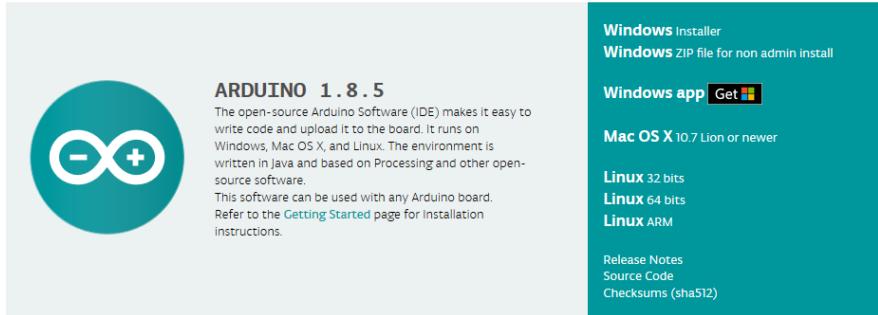
- Install Arduino IDE
- Install Git SCM
- Git GUI, source, target > clone
- Double click get.exe

 ดาวน์โหลดและติดตั้งไฟล์ที่

<https://www.arduino.cc/en/Main/Software>

HOME BUY SOFTWARE PRODUCTS LEARNING COMMUNITY SUPPORT

Download the Arduino IDE



ARDUINO 1.8.5
The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. It runs on Windows, Mac OS X, and Linux. The environment is written in Java and based on Processing and other open-source software.
This software can be used with any Arduino board. Refer to the [Getting Started](#) page for installation instructions.

Windows Installer
Windows ZIP file for non admin install

Windows app Get 

Mac OS X 10.7 Lion or newer

Linux 32 bits
Linux 64 bits
Linux ARM

Release Notes
Source Code
Checksums (sha512)

 ดาวน์โหลดและติดตั้งไฟล์ที่

<https://git-scm.com/>



About
The advantages of Git compared to other source control systems.

Documentation
Command reference pages, Pro Git book content, videos and other material.

Downloads
GUI clients and binary releases for all major platforms.

Community
Get involved! Bug reporting, mailing list, chat, development and more.

Pro Git by Scott Chacon and Ben Straub is available to [read online for free](#). Dead tree versions are available on [Amazon.com](#).

Latest source Release
2.12.2
[Release Notes \(2017-03-24\)](#)

Downloads for Windows

 Windows GUIs  Tarballs

 Mac Build  Source Code



Git Gui

Clone Existing Repository

Source Location: <https://github.com/espressif/arduino-esp32.git>

Target Directory: C:/admin/Documents/Arduino/hardware/espressif/esp32

Clone Type:

- Standard (Fast, Semi-Redundant, Hardlinks)
- Full Copy (Slower, Redundant Backup)
- Shared (Fastest, Not Recommended, No Backup)

Recursively clone submodules too

- Git Gui
- Select Clone Existing Repository
- Source: <https://github.com/espressif/arduino-esp32.git>
- Target: C:/Users/[YOUR_USER_NAME]/Documents/Arduino/hardware/espressif/esp32
- Click Clone
- Open C:/Users/[YOUR_USER_NAME]/Documents/Arduino/hardware/espressif/esp32/tools and double-click get.exe



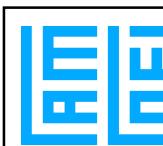
updated

Git CMD

```
C:\Users\admin>cd C:\Users\admin\Documents\Arduino\hardware\espressif\esp32
C:\Users\admin\Documents\Arduino\hardware\espressif\esp32>git pull origin master
From https://github.com/espressif/arduino-esp32
 * branch            master      -> FETCH_HEAD
Already up-to-date.

C:\Users\admin\Documents\Arduino\hardware\espressif\esp32>
```

- Git CMD
- cd C:/Users/[YOUR_USER_NAME]/Documents/Arduino/hardware/espressif/esp32
- git pull origin master



ไฟลเดอร์

- งานเอกสาร Arduino

C:\Users\[YOUR_USER_NAME]\Documents\Arduino

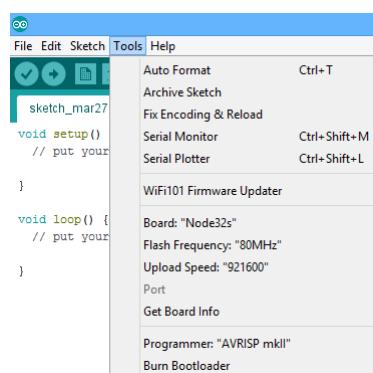
- ไฟลเดอร์ Arduino IDE

C:\Program Files (x86)\Arduino

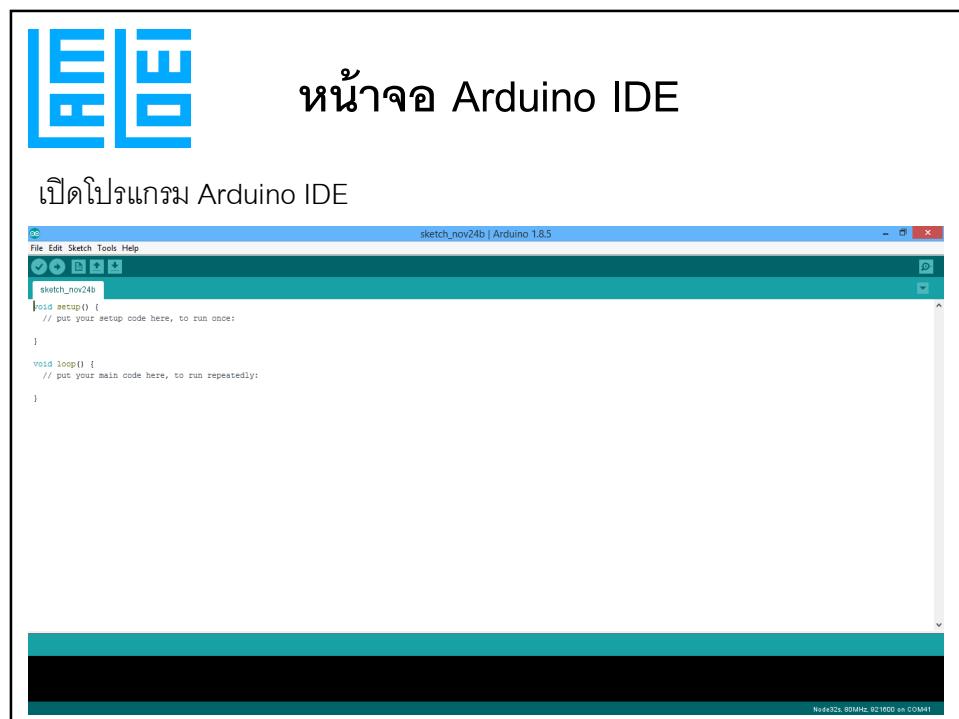
C:\Users\[YOUR_USER_NAME]\AppData\Local\Arduino15

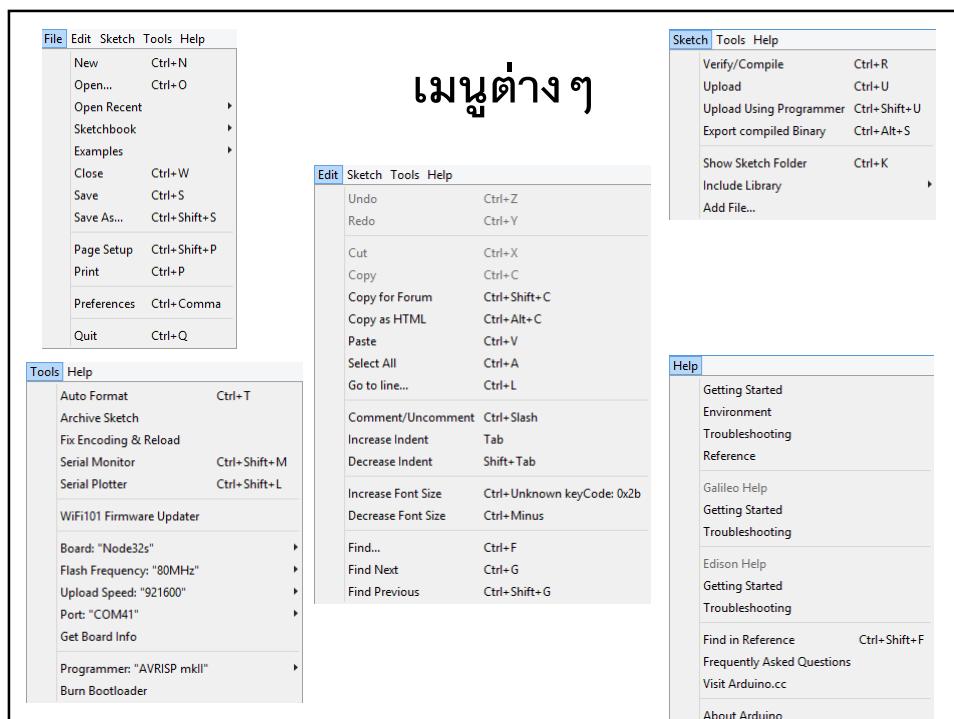
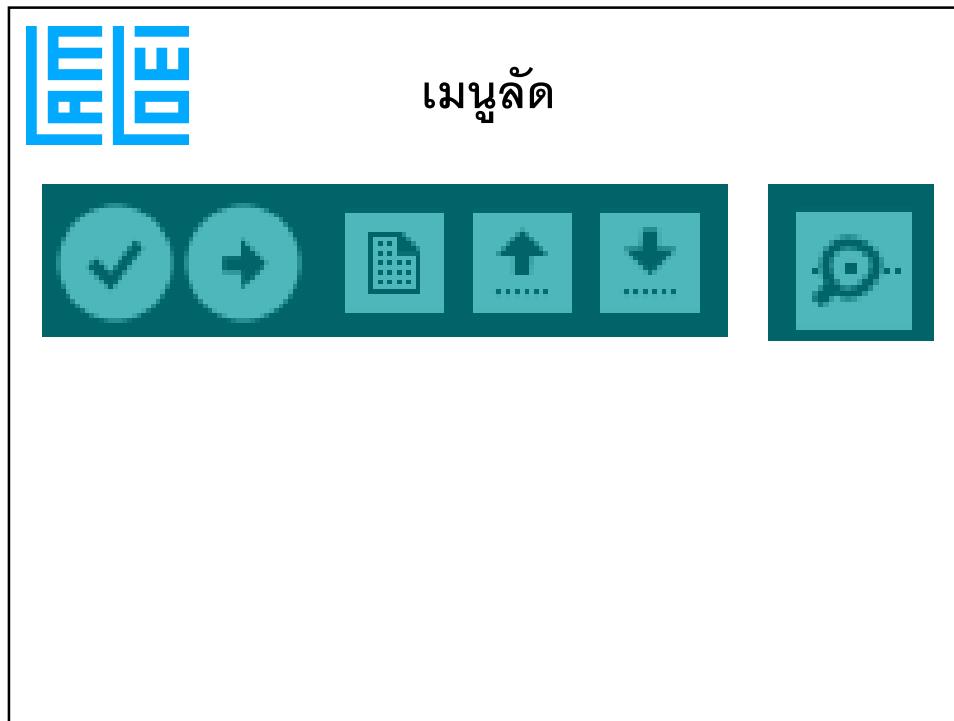


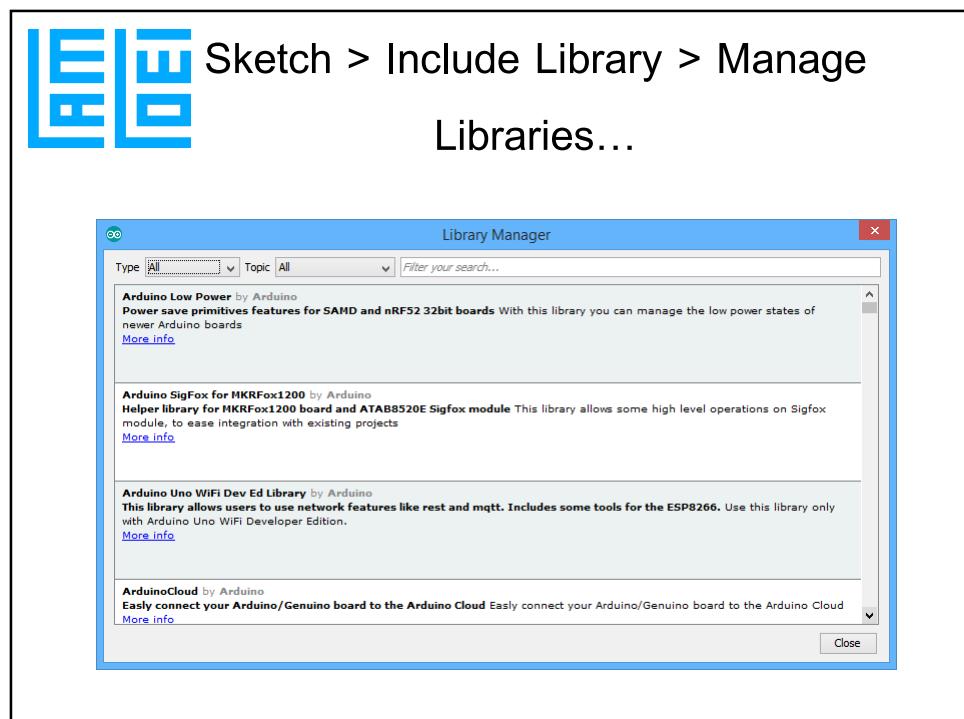
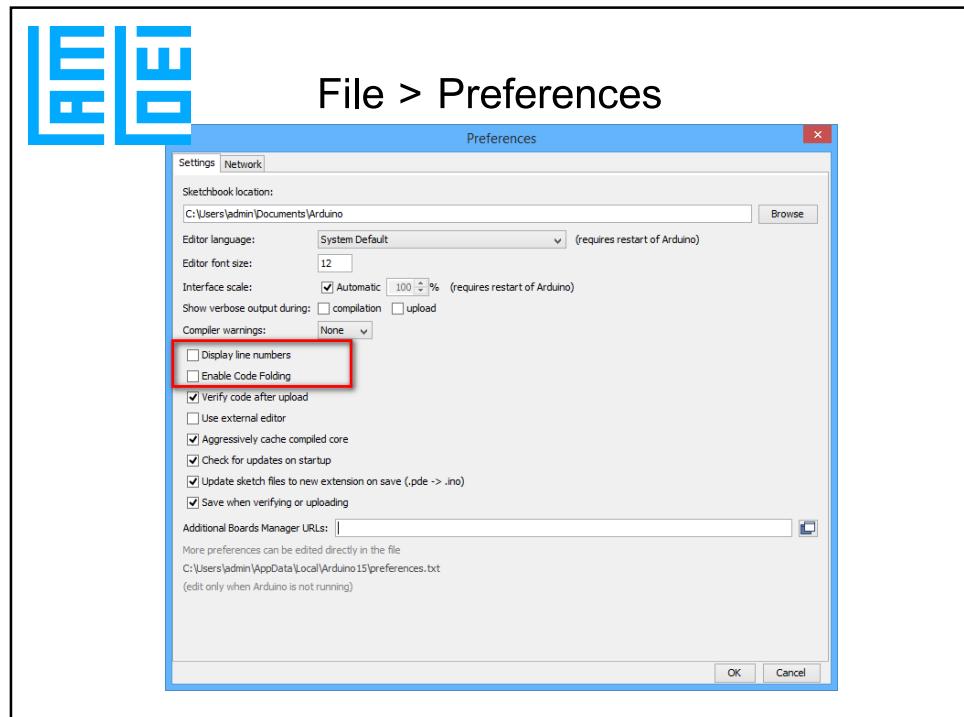
Tools > Board: “Node32s”

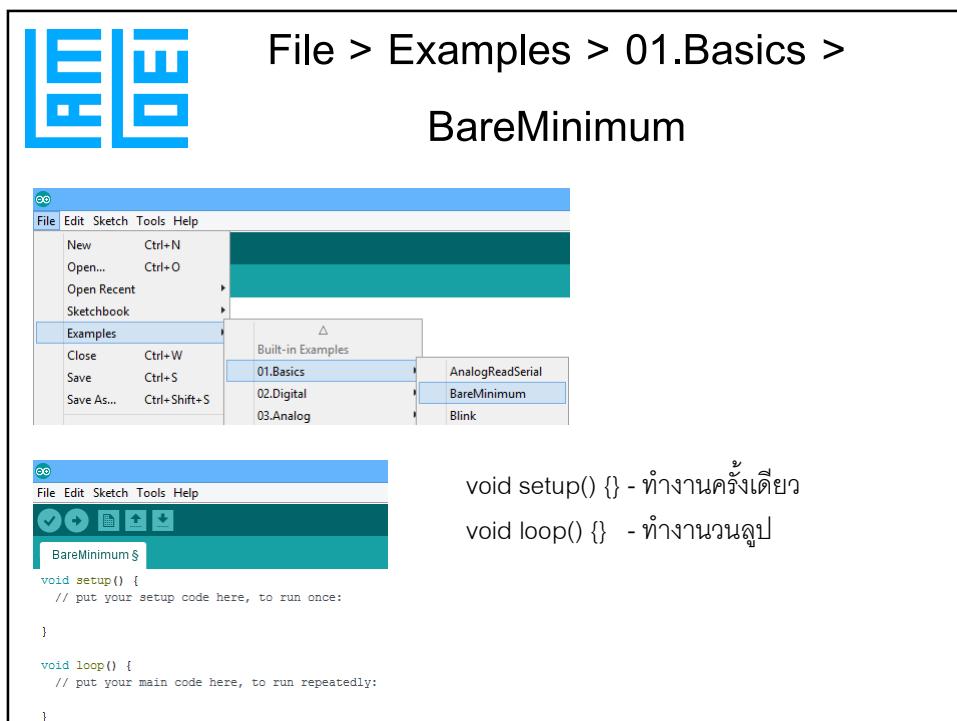
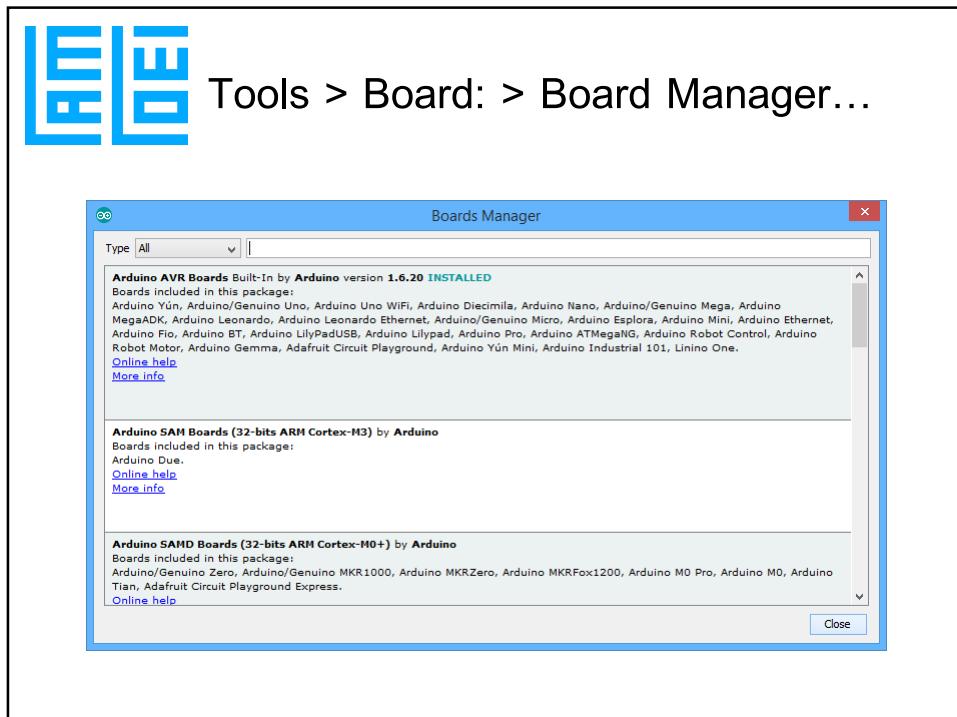


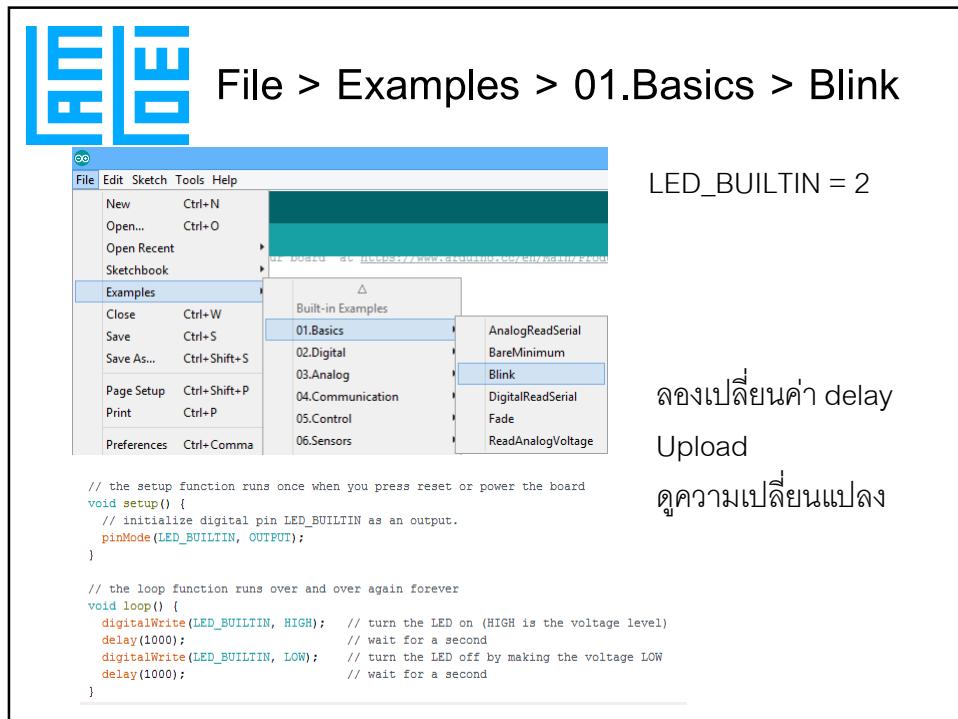
จะต้องเห็นทั้ง Board และ Port

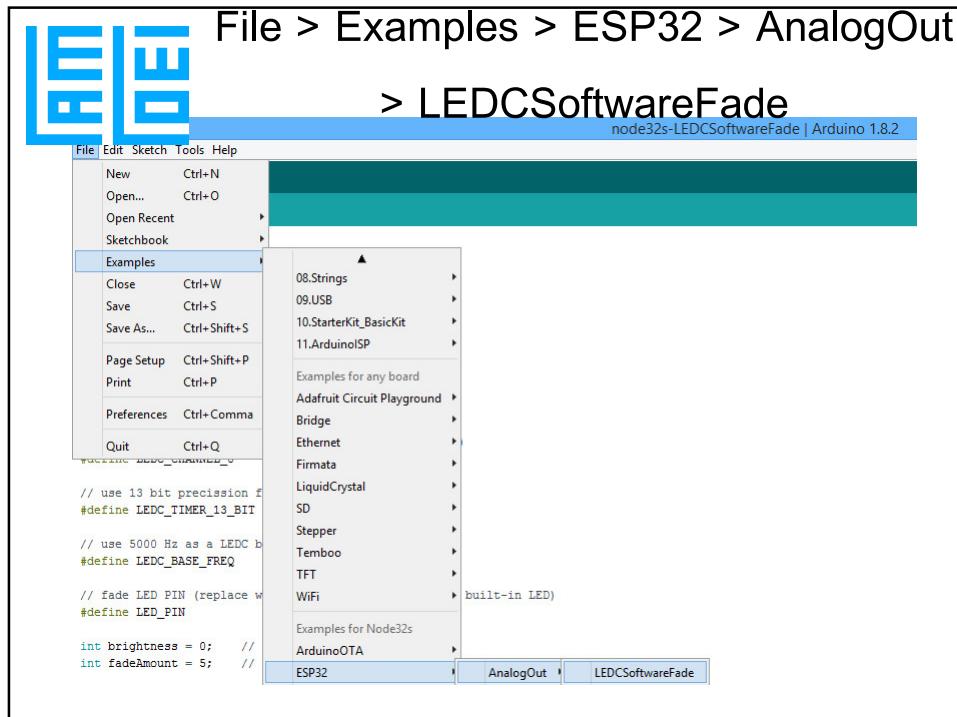












The screenshot shows the Arduino IDE after the sketch has been uploaded. The title bar reads "node32s-LEDCSoftwareFade | Ar". The code editor window shows the same code as above, with the line "#define LED_PIN 2" highlighted by a red box. The status bar at the bottom of the IDE window displays the following messages: "Done uploading.", "Hash of data verified.", "Leaving...", and "Board resetting...".

```

#define LEDC_TIMER_13_BIT 13

#define LEDC_BASE_FREQ 15000

#define LED_PIN 2

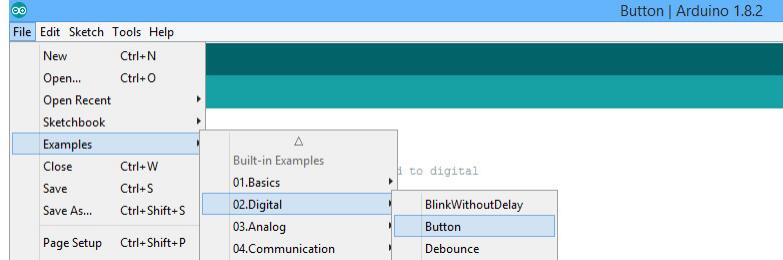
int brightness = 0; // how bright the LED is
int fadeAmount = 5; // how many points to fade the LED by

```

#define LEDC_BASE_FREQ 15000
#define LED_PIN 2



File > Examples > 02.Digital > Button



```

const int buttonPin = 0;
const int ledPin = 2;

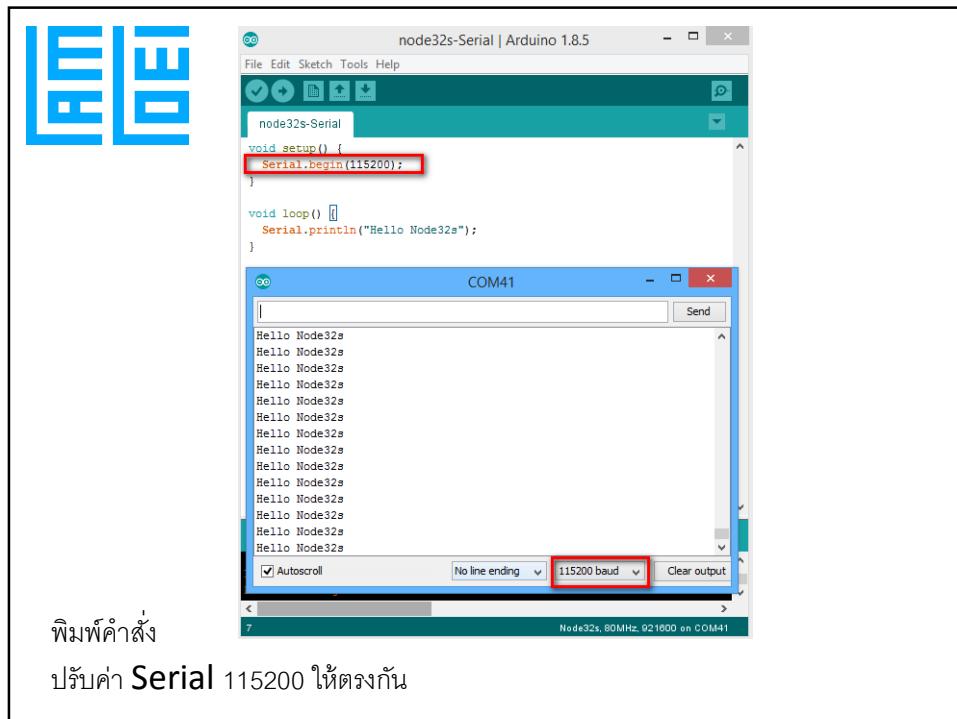
```

กดปุ่มไฟสว่าง ปล่อยปุ่มไฟดับ



แบบฝึกหัด (15นาที)

- ให้ กดปุ่มไฟดับ ปล่อยปุ่มไฟสว่าง



ស្រុបខ័ណ្ឌ

- File > Examples
 - ปรับค่า
 - Upload Program
 - สังเกตความเปลี่ยนแปลง



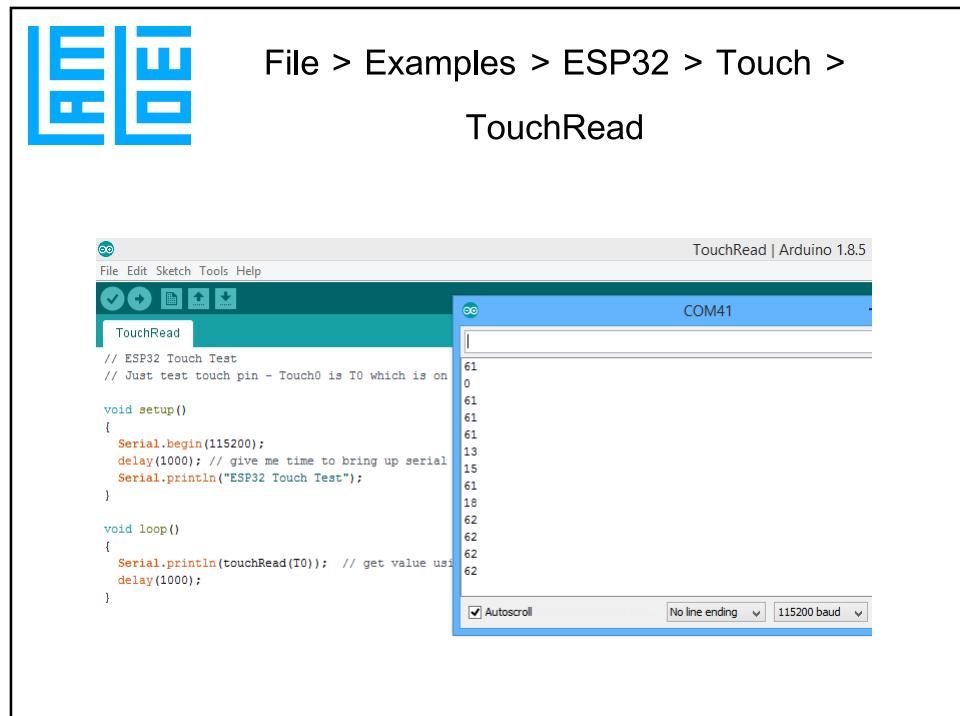
แบบฝึกหัด 30 นาที

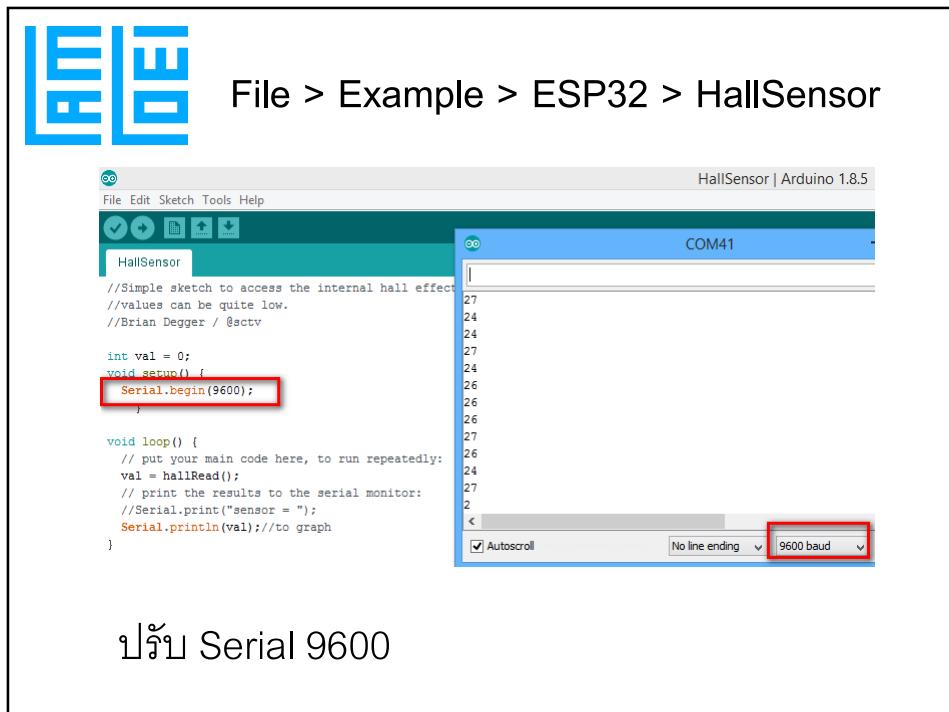
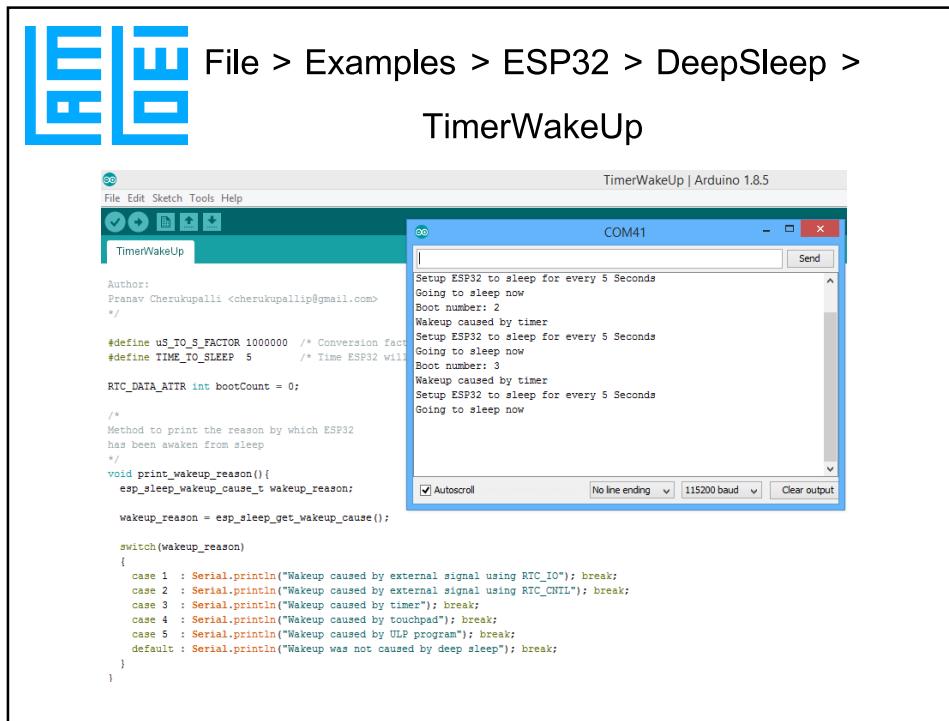
- เลือก File > Example
- ปรับค่า และสังเกตความเปลี่ยนแปลง
- จงอธิบายตัวอย่างนั้น

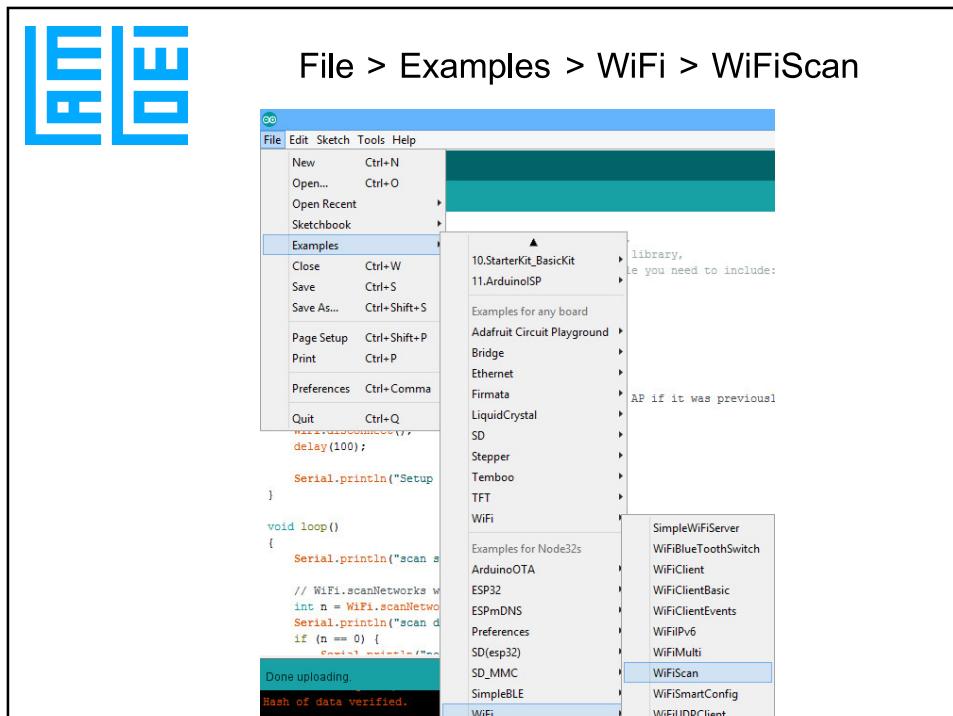
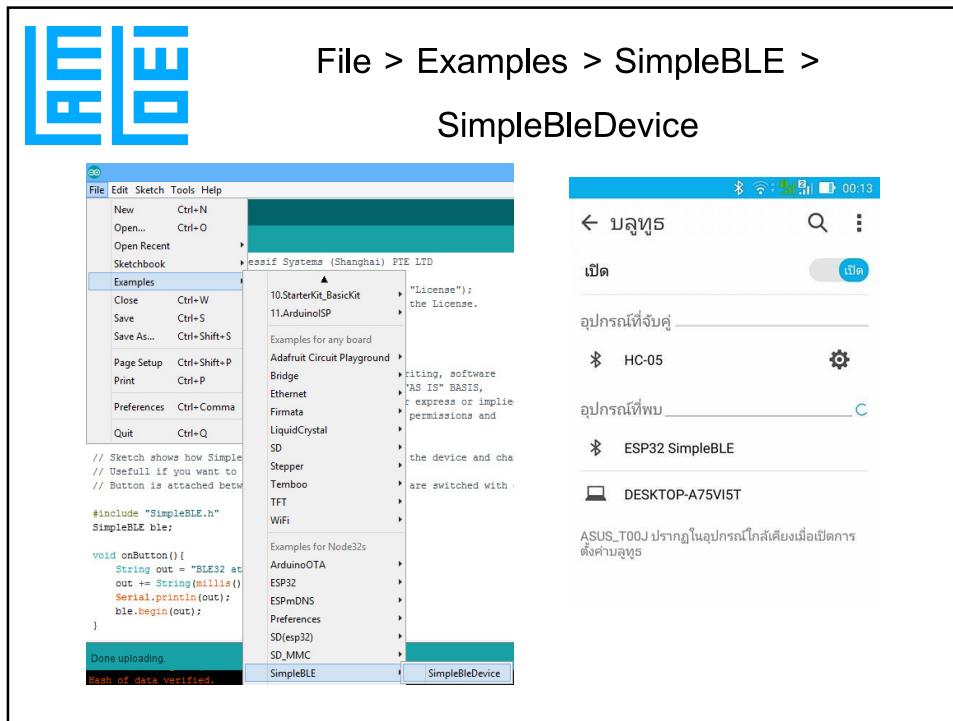


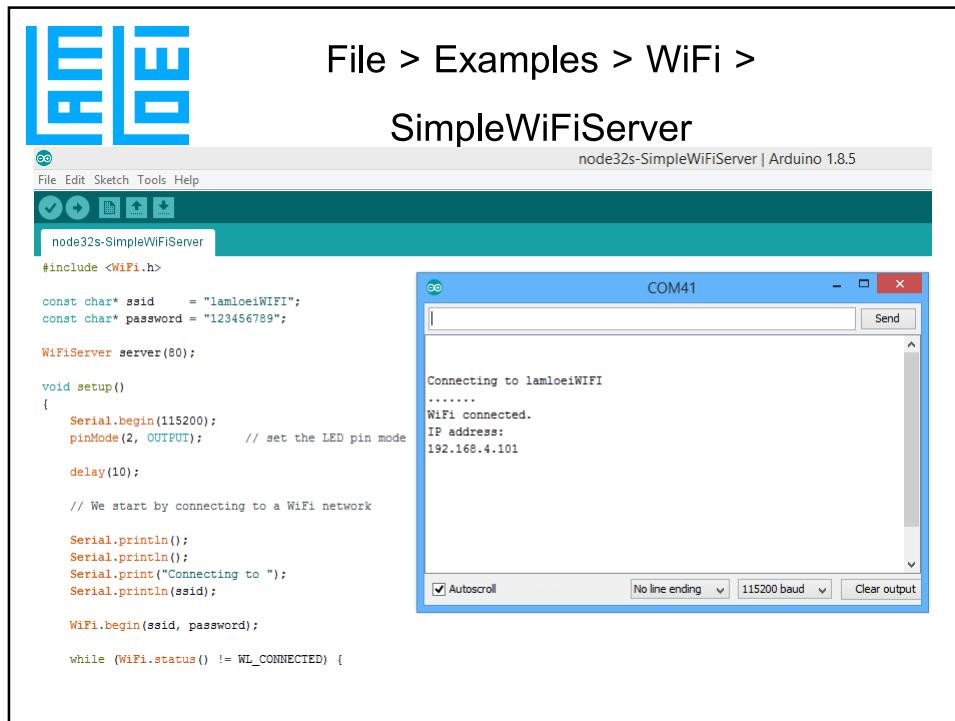
Lunch Break

- 60 นาที









const char* ssid = "lamloeiWIFI";
 const char* password = "123456789";

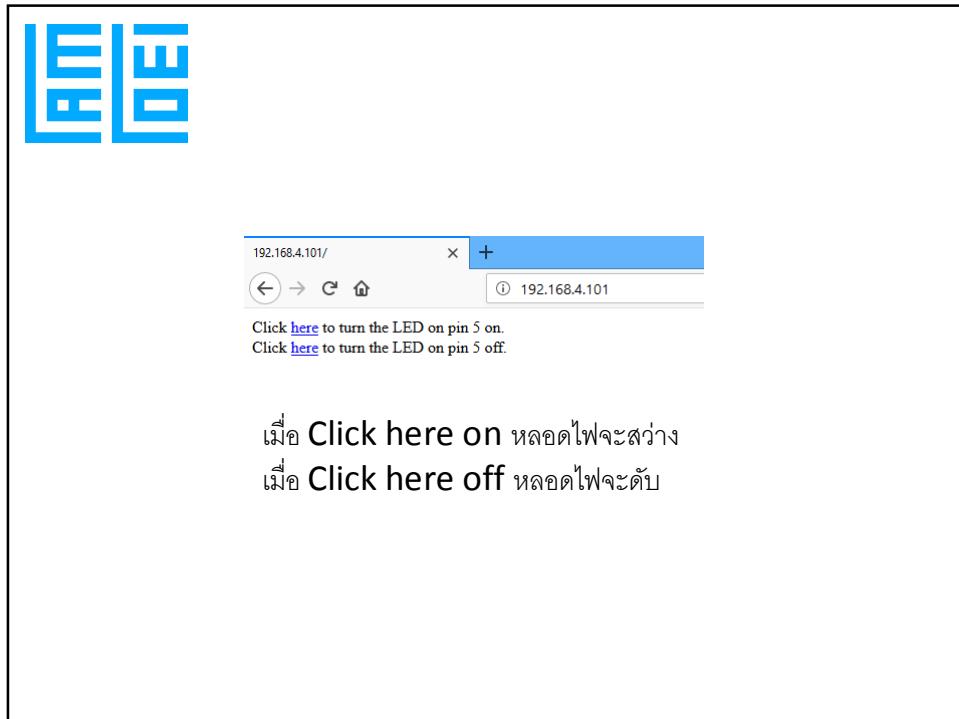
```

pinMode(2, OUTPUT);

if (currentLine.endsWith("GET /H")) {
  digitalWrite(2, LOW);           // GET /H turns the LED on
}

if (currentLine.endsWith("GET /L")) {
  digitalWrite(2, HIGH);          // GET /L turns the LED off
}

```



เพิ่ม Code จากภายนอก

เปิดเบราว์เซอร์ไปที่ <https://playground.arduino.cc/Main/I2cScanner>

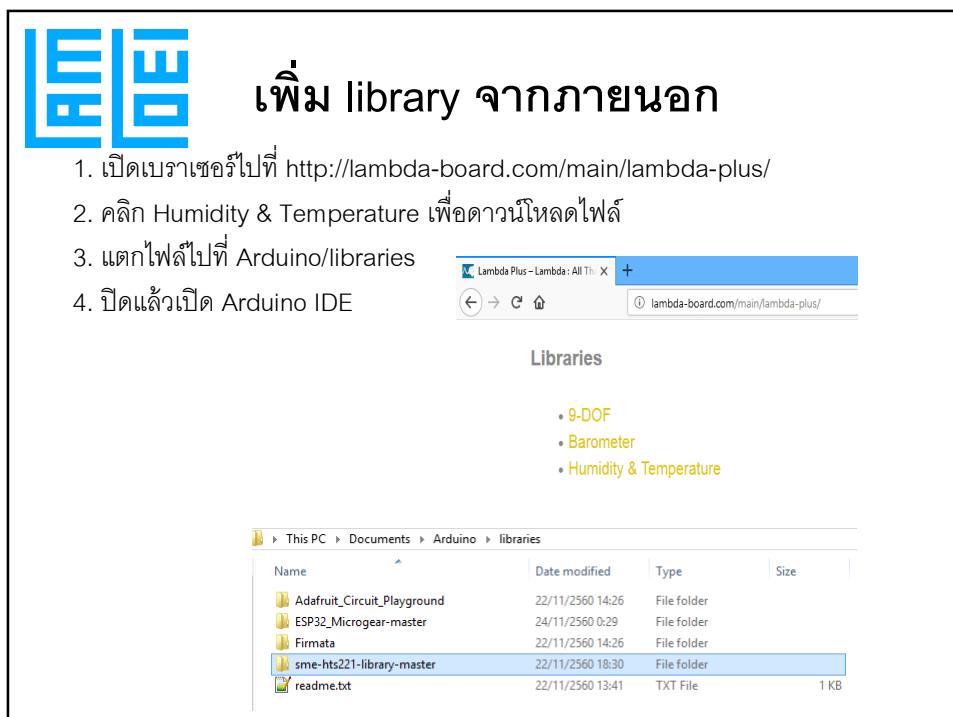
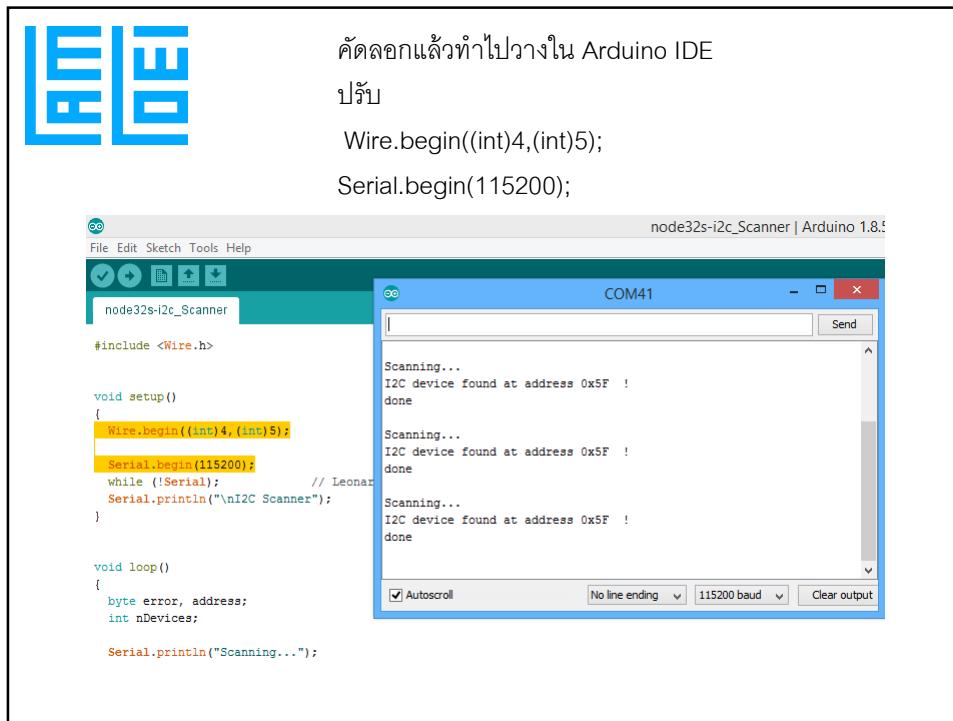
คลิก Get Code

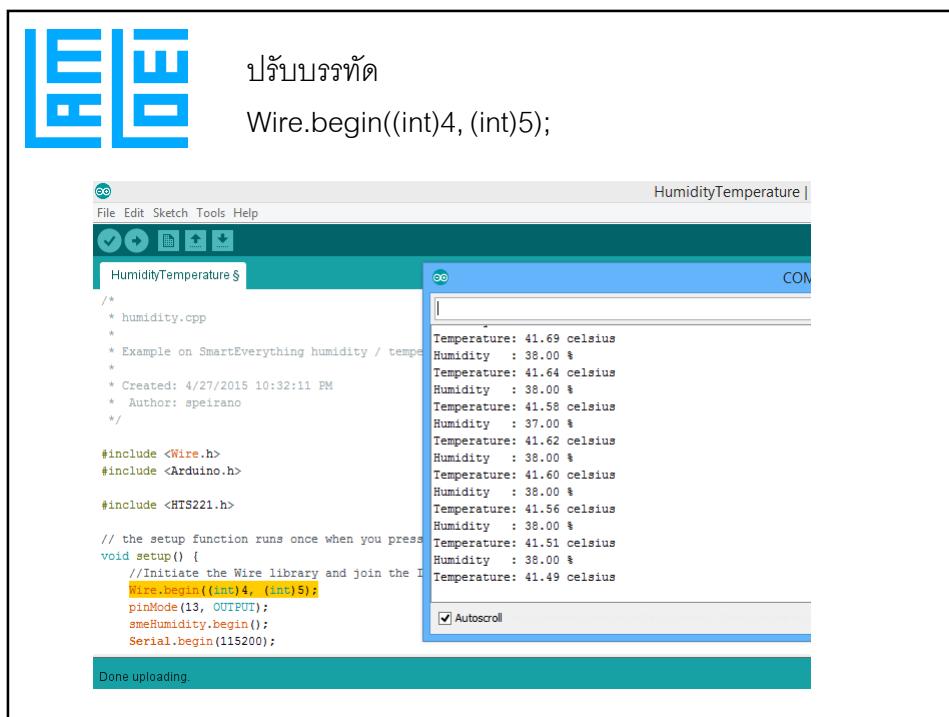
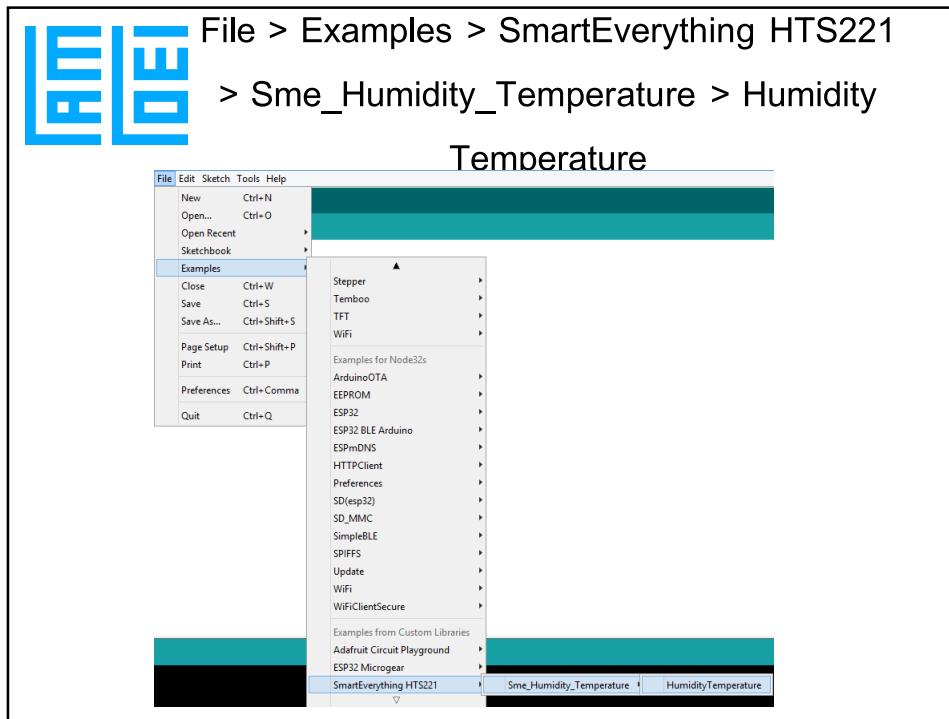
<https://playground.arduino.cc/Main/I2cScanner>

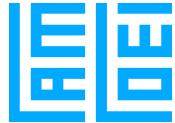
```

73.     Serial.print("0");
74.     Serial.println(address,HEX);
75.   }
76. }
77. if (nDevices == 0)
78.   Serial.println("No I2C devices found\n");
79. else
80.   Serial.println("done\n");
81.
82. delay(5000);      // wait 5 seconds for next scan
83. }
```

[\[Get Code\]](#)







Break

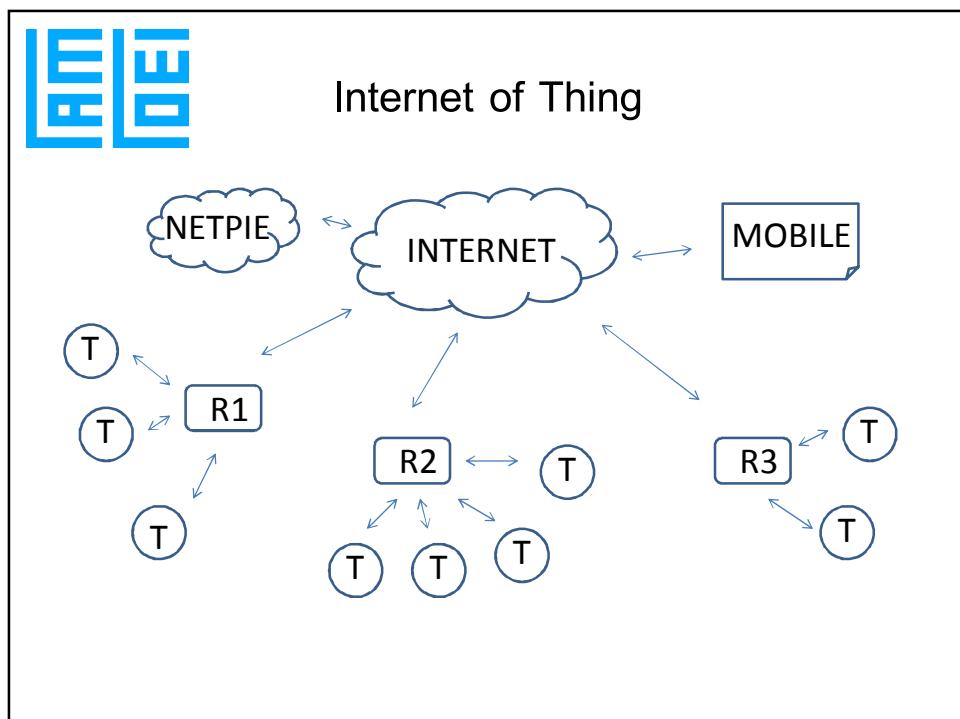
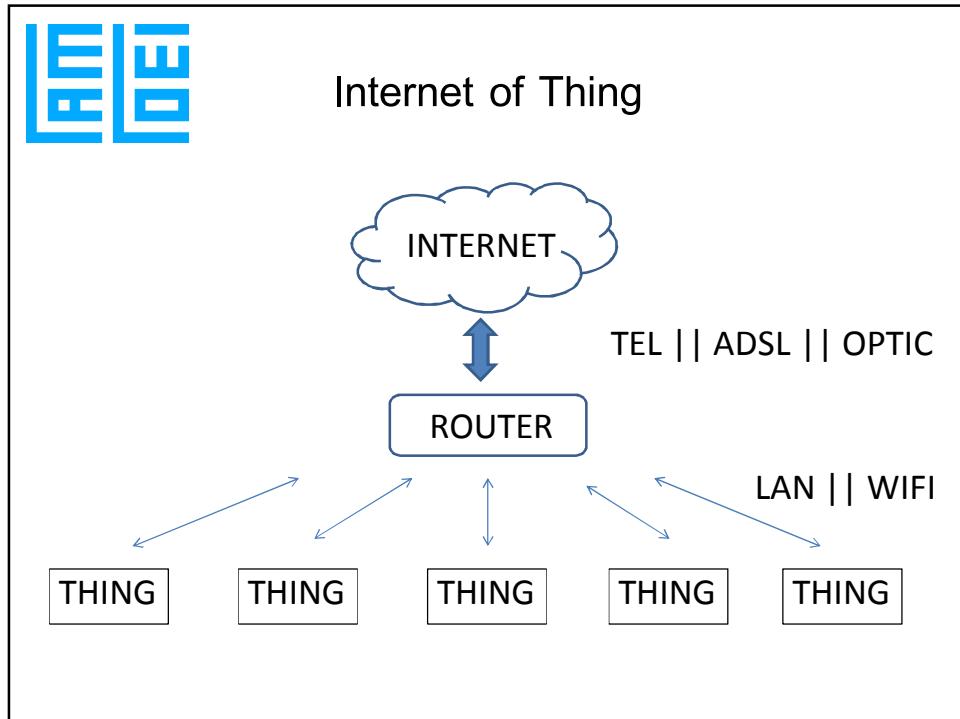
- 15 นาที



สมัคร Netpie

จะได้

1. User
2. Password - สำหรับเข้าหน้าเว็บ
3. Appid
4. Appkey
5. Appsecret - สำหรับให้ Thing เข้าใช้งาน



สมัคร Netpie

<https://netpie.io/>

SIGN UP FREE ແລ້ວທຳຕາມຂັ້ນຕອນ

LOGIN

USERNAME OR EMAIL ADDRESS
lamloei

PASSWORD

LOGIN

Forgot your password?

The screenshot shows the NETPIE platform interface under the 'RESOURCES > APPLICATIONS +' section. At the top, there's a navigation bar with links for HOME, PARTNERS, DEVELOPERS, BLOG, and RESOURCES. Below the navigation is a search bar labeled 'Google Custom Search'. The main content area displays a summary: '1 APPLICATION' and '0 THINGS'. A sidebar on the right lists 'APPLICATIONS', 'FREEBOARDS', and 'FEEDS(beta)'. In the center, there's a modal window for creating a new application. The modal has a title 'APPLICATION', a preview section showing a small icon for 'laml', and a text input field containing 'LAML'. At the bottom of the modal are 'CREATE' and 'CANCEL' buttons. A red circle highlights the '+' button in the top right corner of the modal.

This screenshot shows the 'APPLICATION (APPID)' creation page. The top part of the page displays an application entry for 'LAML' with a delete button. To the right, there's a section for 'APPLICATION KEY' showing '0' entries and a search icon. A red circle highlights the '+' button in the top right corner of the application key section. Below this, there's a form for creating a new application key, with fields for 'MyDevice' and 'Device Key'. At the bottom are 'CREATE' and 'CANCEL' buttons. A red circle also highlights the '+' button in the top right corner of this form. The NETPIE logo is visible in the bottom right corner of the page.

NETPIE
make things smart

MyDevice

Key : LAML

Secret : XXX

REST API auth : XXX

RENAM CANCEL

DELETE

APPLICATION KEY

Device Key, Session Key

APPLICATION KEY

MyDevice [online 0]

จะได้

APPID: LAML

Key: XXX

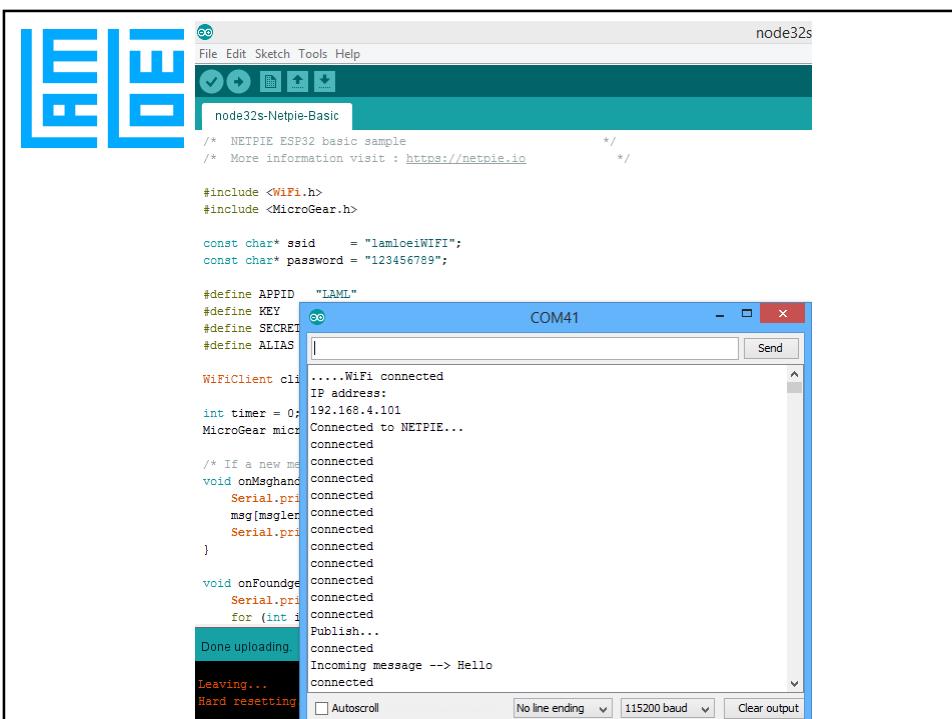
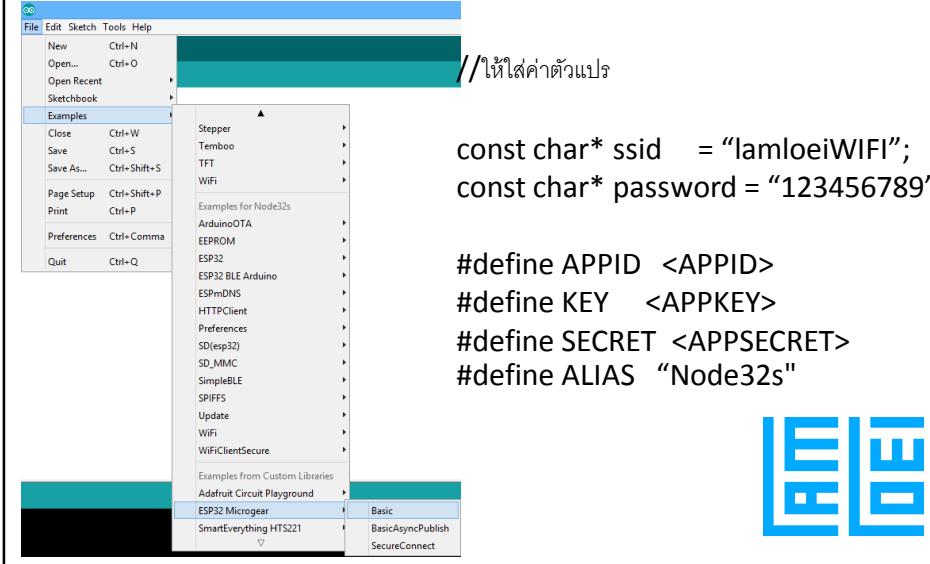
Secret: XXX

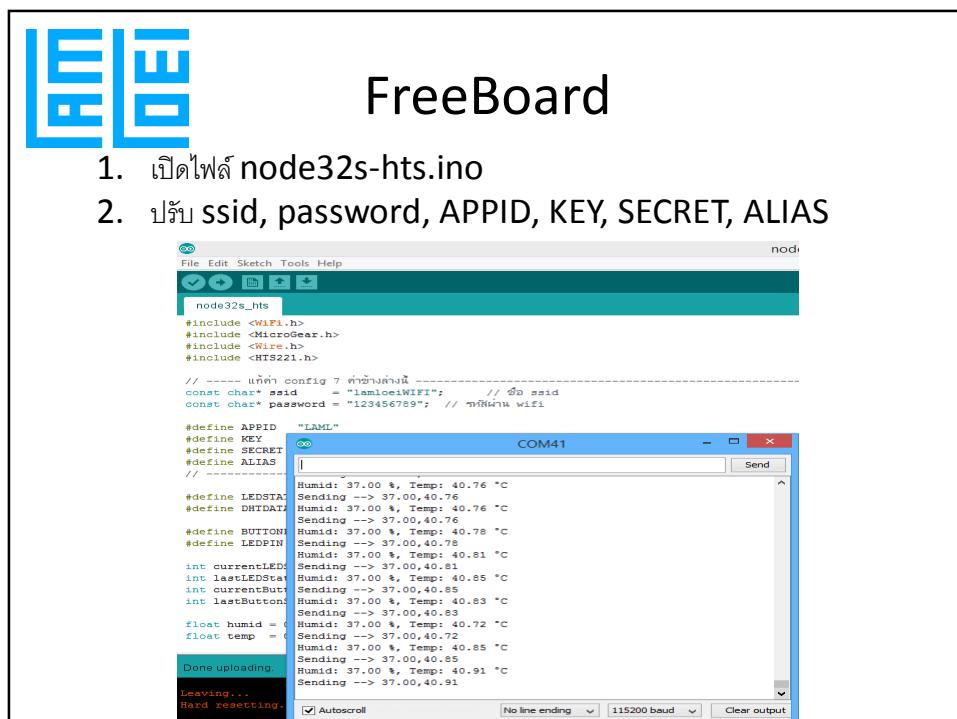
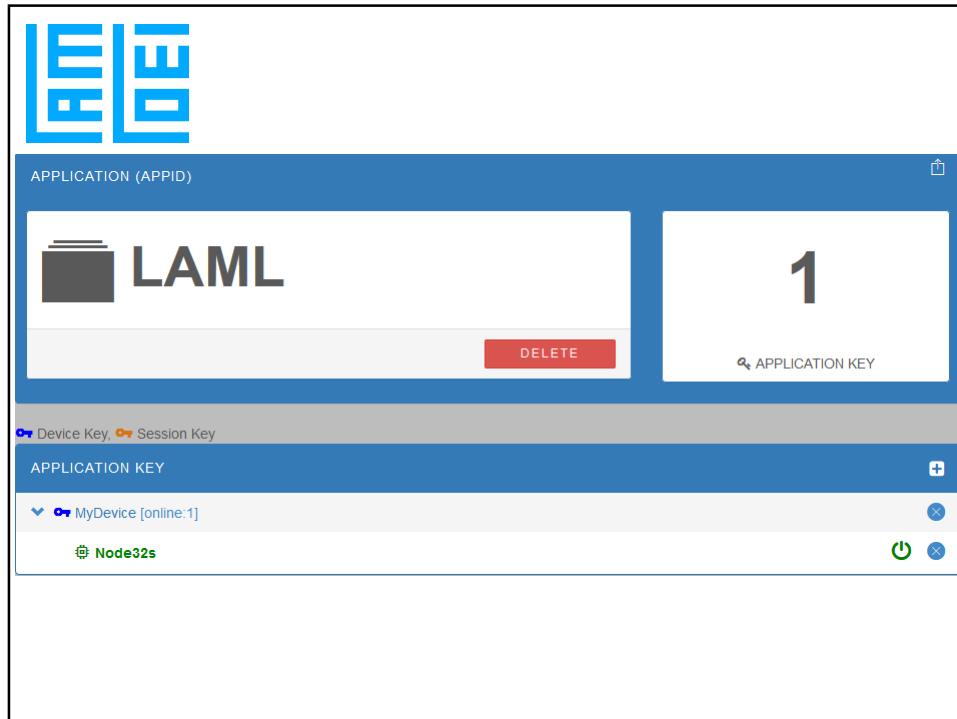
LAME

ดาวน์โหลด Library

- ดาวน์โหลดไฟล์ https://github.com/lamloei/ESP32_Microgear
- แตกไฟล์ลงโฟลเดอร์ Arduino/libraries
- ปิดและเปิด Arduino IDE ใหม่

File > Examples > ESP32 Microgear > Basic





The screenshot shows the LAML application management interface. At the top, there is a logo consisting of the letters 'LAML' in blue and white. Below the logo, a header bar has the text 'APPLICATION (APPID)' on the left and a trash can icon on the right. The main content area contains two large boxes: one on the left labeled 'LAML' with a folder icon and a 'DELETE' button, and one on the right labeled '1' with a search icon and the text 'APPLICATION KEY'. Below these boxes is a section titled 'Device Key, Session Key'. Underneath, a table lists 'MyDevice [online:2]' with two rows: 'laml_Freeboard' and 'myFeed'. The 'myFeed' row is highlighted with a red border.

The screenshot shows the NETPIE Resources > Freeboards interface. At the top, there is a logo for 'NETPIE where things chat'. Below the logo, a header bar has the text 'RESOURCES > FREEBOARDS' on the right. The main content area contains a table with three columns: 'APPLICATION (APPID)', 'FREEBOARDS', and 'FEEDS'. The 'FREEBOARDS' column has a single entry labeled 'laml_Freeboard'. Below the table, there is a large button labeled 'CLICK ADD'. At the bottom, there is a sidebar for 'NETPIE Freeboard' with options: 'IMPORT', 'EXPORT', 'RESET', '+ ADD PANE', and a 'DATA SOURCES' section. In the 'DATA SOURCES' section, there is a table with a single row for 'laml_Freeboard' with a 'Last Updated' timestamp of '1:25:18 AM'. A red box highlights the 'ADD' button in the 'DATA SOURCES' section.



DATA SOURCE

Connect to NETPIE as a microgear to communicate real-time with other microgears in the same App ID. The microdatasource is referenced by microgear[DATASOURCENAME]

TYPE	NETPIE Microgear
NAME	
APP ID	
NETPIE App ID obtained from https://netpie.io/app	
KEY	
Key	
SECRET	
Secret	
SUBSCRIBED TOPICS	/#
Topics of the messages that this datasource will consume, the default is /# which means all messages in this app ID.	
ONCREATED ACTION	
JS code to run after a datasource is created	
ONCONNECTED ACTION	
JS code to run after a microgear datasource is connected to NETPIE	
SAVE CANCEL	



- TYPE NETPIE Mirogear
- ตั้งชื่อ NAME
- ใส่ค่า APP ID
- KEY
- SECRET
- แล้วคลิกปุ่ม SAVE

ถ้าเขียนต่อได้จะขึ้นชื่อ Name ที่ตั้งไว้ กับค่า Last Updated

ADD PANE > + 

TYPE เป็น Text
 value เป็น datasources["laml_Freeboard"]["/LAML/dht/myFeed"]
 คลิกปุ่ม SAVE

WIDGET

TYPE	Text
TITLE	<input type="text"/>
SIZE	Regular
VALUE	<input type="text" value="datasources['laml_Freeboard']['/LAML/dht/myFeed']"/> + DATA SOURCE ✖ JS EDITOR
INCLUDE SPARKLINE	<input checked="" type="checkbox"/> NO
ANIMATE VALUE CHANGES	<input checked="" type="checkbox"/> YES <input type="checkbox"/>
UNITS	<input type="text"/>
SAVE CANCEL	

ADD PANE > + 

TYPE เป็น Gauge
 TITLE เป็น อุณหภูมิ
 value เป็น datasources["laml_Freeboard"]["/LAML/dht/myFeed"].split(",")[0]
 คลิกปุ่ม SAVE

WIDGET

TYPE	Gauge
TITLE	อุณหภูมิ
VALUE	<input type="text" value="0"/> < > + DATA SOURCE ✖ JS EDITOR
UNITS	<input type="text"/>
MINIMUM	0
MAXIMUM	100
SAVE CANCEL	

ADD PANE > +

TYPE เป็น Gauge

TITLE เป็น ความชื้น

value เป็น datasources["laml_Freeboard"]["/LAML/dht/myFeed"].split(",")[1]

คลิกปุ่ม SAVE

The screenshot shows the 'Widget' configuration screen. The 'TYPE' dropdown is set to 'Gauge'. The 'TITLE' field contains the Thai text 'ความชื้น'. The 'VALUE' field has a small input box with a left arrow and right arrow, indicating it's a data source. To its right are buttons for '+ DATA SOURCE' and 'JS EDITOR'. Below these are fields for 'UNITS', 'MINIMUM' (set to 0), and 'MAXIMUM' (set to 100). At the bottom are 'SAVE' and 'CANCEL' buttons.

Freeboard

The screenshot shows the main Freeboard dashboard. On the left, there's a sidebar with 'NETPIE Freeboard' and buttons for 'IMPORT', 'EXPORT', 'RESET', and '+ ADD PANE'. On the right, under 'DATA SOURCES', there's a table with one entry: 'Name: laml_Freeboard, Last Updated: 1:41:21 AM'. Below the table are 'ADD' and 'DELETE' buttons. The main area features three cards. The first card shows the text '36.00,42.58'. The second card is labeled 'อุณหภูมิ' and has a gauge with the value '36'. The third card is labeled 'ความชื้น' and has a gauge with the value '42.58'.



THE END

- Q & A

