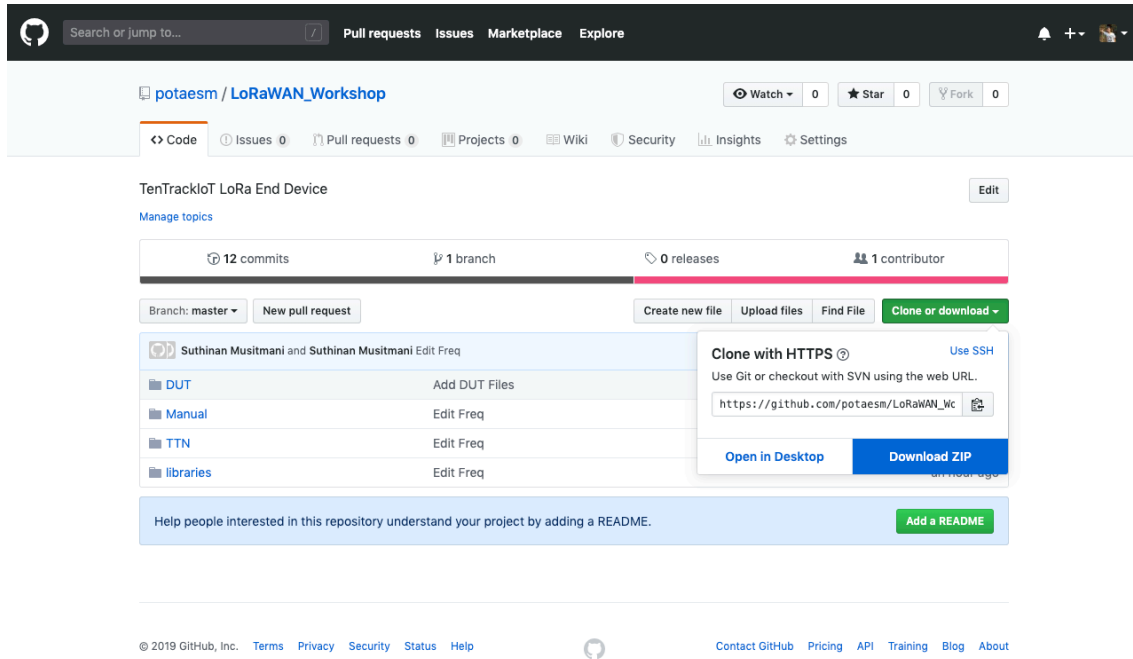
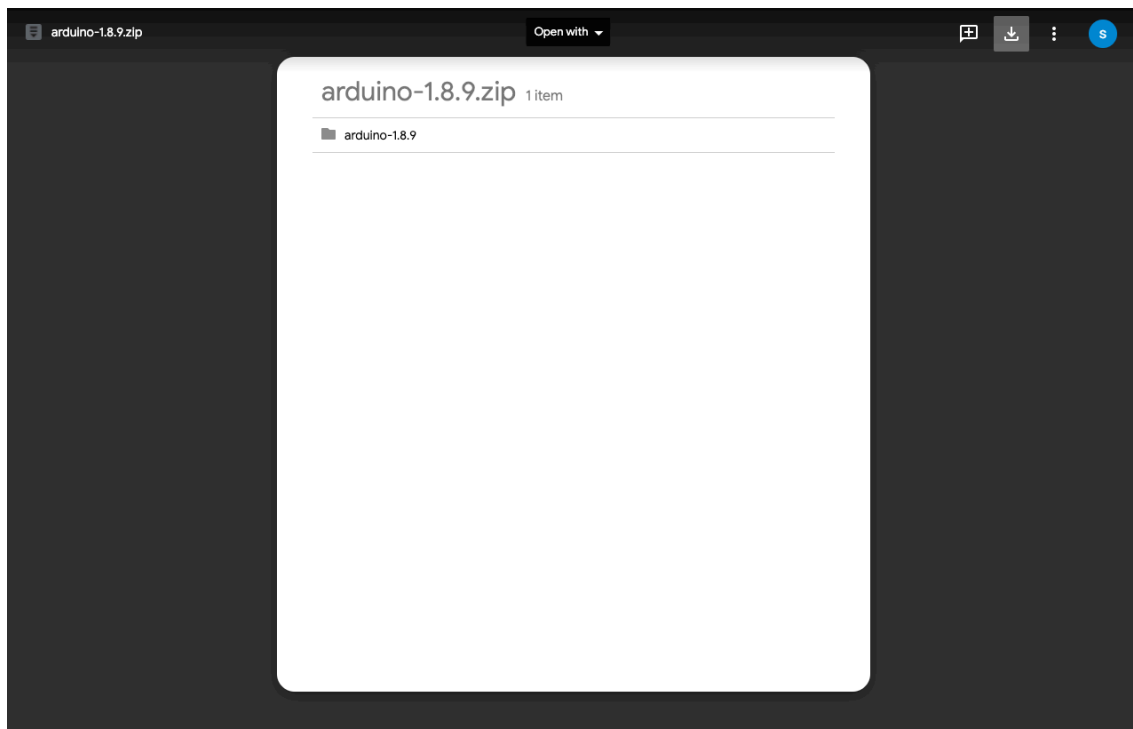


ขั้นตอนการดาวน์โหลด ติดตั้งโปรแกรม และการปรับเปลี่ยนค่าของตัวแปรภายในโปรแกรม

1. ดาวน์โหลดไฟล์ไลบรารี โปรแกรมและคู่มือได้จาก https://github.com/potaesm/LoRaWAN_Workshop โดยกดปุ่ม Clone or download จากนั้นกดปุ่ม Download ZIP ดังรูปที่ 1 และดาวน์โหลดโปรแกรม Arduino IDE สำหรับใช้ในการเขียนโปรแกรม และอัปเดตไปยังอุปกรณ์ได้จาก <http://bit.do/189IDE> ดังรูปที่ 2



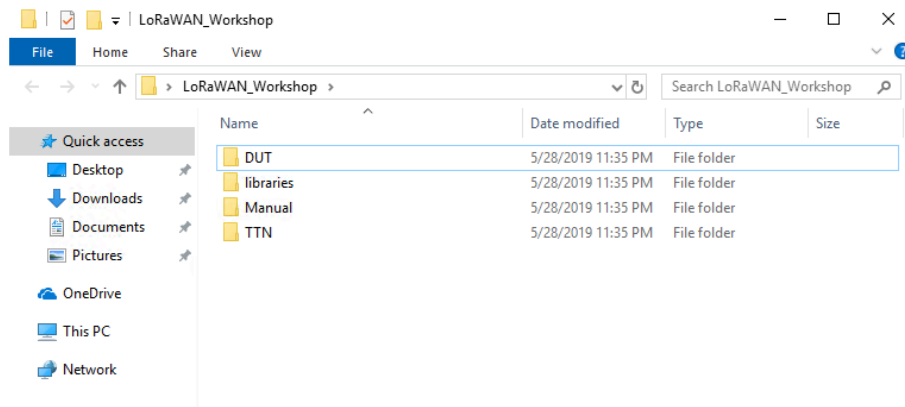
รูปที่ 1 การดาวน์โหลดไฟล์ไลบรารี โปรแกรม และคู่มือ



รูปที่ 2 การดาวน์โหลด Arduino IDE

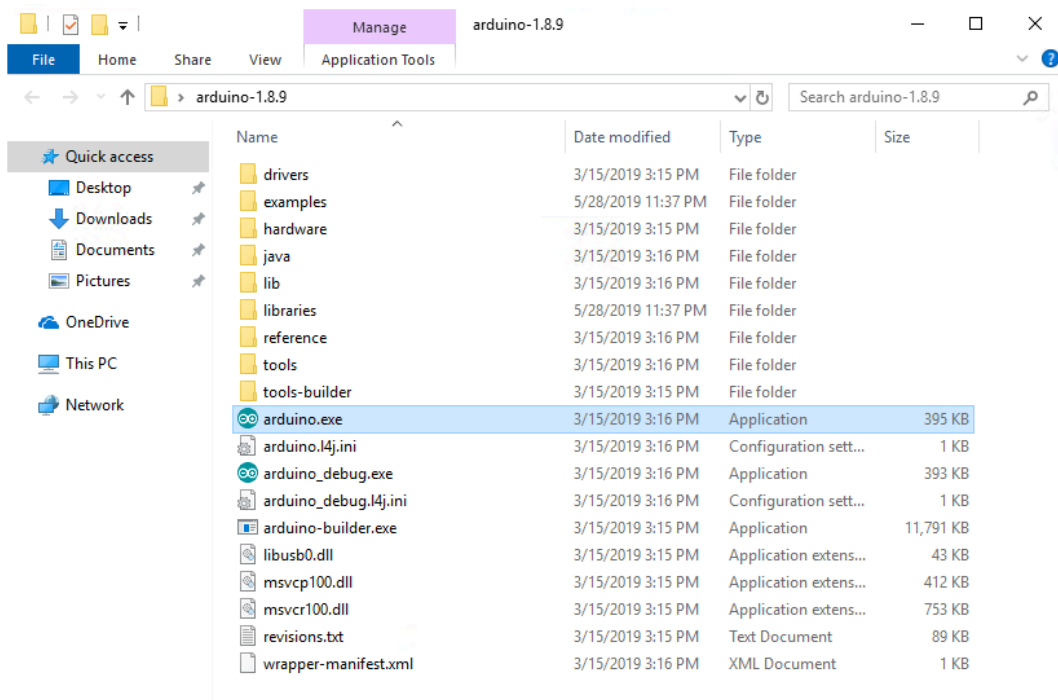
2. เมื่อดาวน์โหลดไฟล์ได้แก่ LoRaWAN_Workshop-master.zip และ arduino-1.8.9.zip เสร็จแล้ว ให้ทำการแตกไฟล์ออกมาจะพบกับไฟล์ต่าง ๆ

- สำหรับไฟล์ LoRaWAN_Workshop-master.zip เมื่อแตกไฟล์แล้วจะได้โฟลเดอร์ที่ประกอบไปด้วย Manual, TTN และ libraries ดังรูปที่ 3 ซึ่งคู่มือนี้จะอยู่ในโฟลเดอร์ Manual ส่วนโปรแกรมทั้งหมดในการ Workshop จะอยู่ในโฟลเดอร์ TTN ซึ่งจะต้องเปิดด้วยโปรแกรม Arduino IDE และสำหรับเครื่องที่มีโปรแกรม Arduino IDE ติดตั้งอยู่แล้ว ให้นำไฟล์ไลบรารีที่อยู่ในโฟลเดอร์ libraries ได้แก่ CayenneLPP, LMIC-Arduino-AS923, SimpleDHT และ UTenTrackIoT ไปวางไว้ที่ Documents/Arduino/libraries/




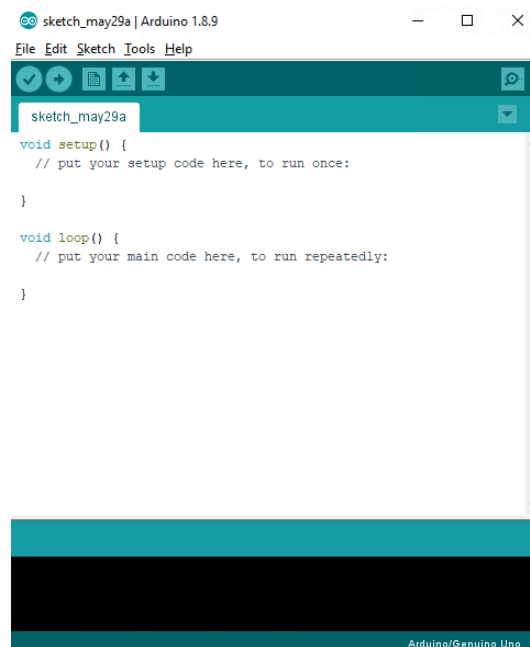
รูปที่ 3 โฟลเดอร์ภายใน LoRaWAN_Workshop-master.zip

- สำหรับไฟล์ arduino-1.8.9.zip เมื่อแตกไฟล์แล้วจะได้โฟลเดอร์ของโปรแกรม Arduino IDE ซึ่งสามารถเปิดได้โดยการดับเบิลคลิกไปที่ arduino.exe ดังรูปที่ 4



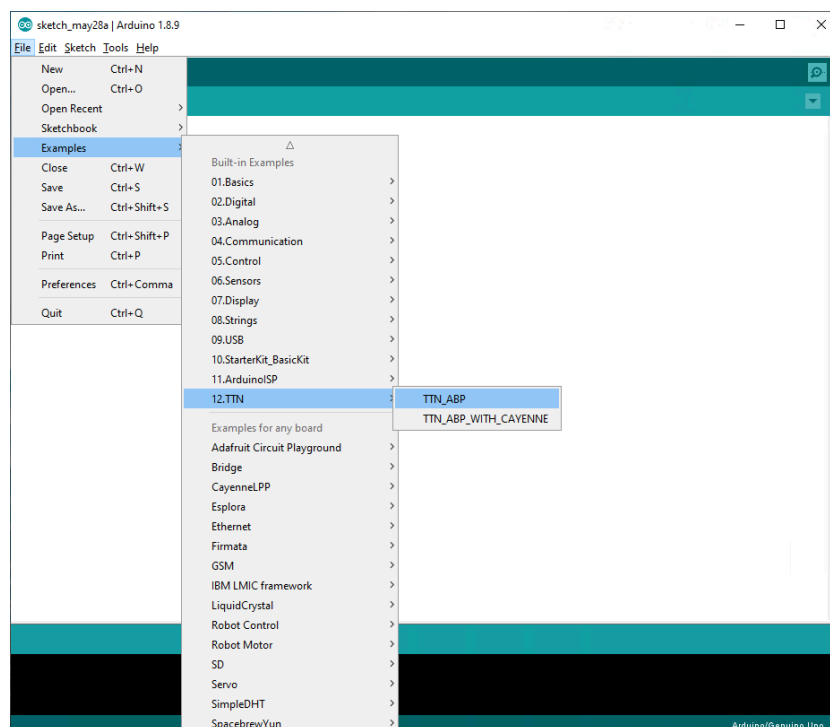
รูปที่ 4 โฟลเดอร์ภายใน arduino-1.8.9.zip

3. เมื่อเปิดโปรแกรมแล้วจะพบกับหน้าต่างโปรแกรม Arduino IDE ดังรูปที่ 5 โดยเมื่อทำการเสียบอุปกรณ์จะต้องปรับ Board และ Port ให้ตรงกับอุปกรณ์นั้น ซึ่งในการ Workshop นี้ใช้ Arduino UNO ส่วน Port สามารถดูได้จาก Device Manager > Ports (COM & LPT) > ชื่ออุปกรณ์ โดยการปรับ Board และ Port นี้จะอยู่ในเมนู Tools > Board และ Tools > Port นอกจากนี้การอัปโหลดโปรแกรมลงบนอุปกรณ์สามารถทำได้โดยการกดปุ่ม 



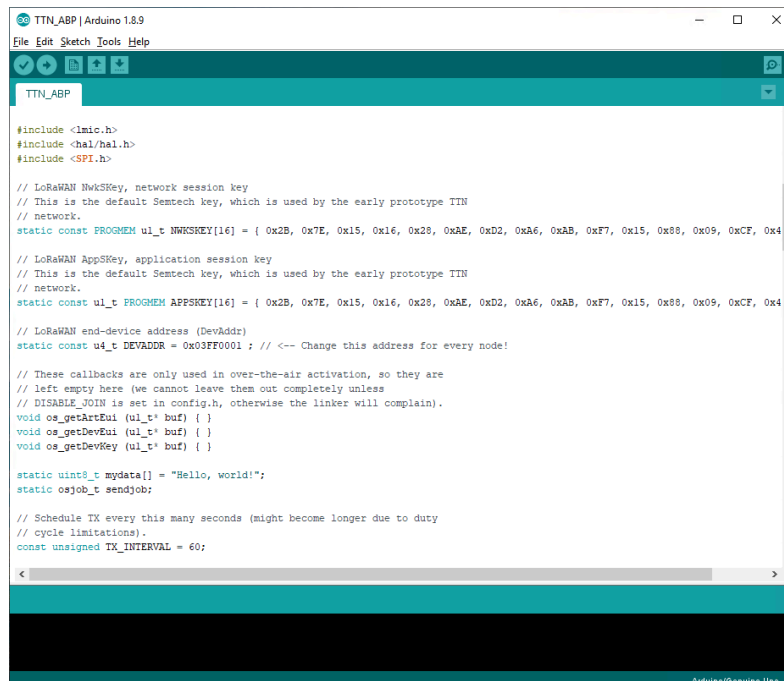
รูปที่ 5 หน้าต่างแรกของโปรแกรม Arduino IDE

4. โปรแกรมที่จะใช้ในการศึกษาภายใน Workshop จะอยู่ทั้งในโฟลเดอร์ TTN ที่แตกออกมาจาก LoRaWAN_Workshop-master.zip ดังกล่าวข้างต้น และนอกจากนี้ยังสามารถเปิดจาก Files > Examples > 12.TTN > ... ดังรูปที่ 6



รูปที่ 6 การเปิดไฟล์โปรแกรมตัวอย่าง

5. สำหรับโปรแกรม TTN_ABP เมื่อเปิดแล้ว จะได้หน้าต่างดังรูปที่ 7 ซึ่งสามารถกำหนด Network session key, App session key และ Device address ได้จากตัวแปร NWKSKEY, APPSKEY และ DEVADDR ตามลำดับ



```
TTN_ABP | Arduino 1.8.9
File Edit Sketch Tools Help

TTN_ABP

#include <lmic.h>
#include <hal/hal.h>
#include <SPI.h>

// LoRaWAN NwksKey, network session key
// This is the default Semtech key, which is used by the early prototype TTN
// network.
static const PROGMEM ui_t NWKSKEY[16] = { 0x2B, 0x7E, 0x15, 0x16, 0x28, 0xAE, 0xD2, 0xA6, 0xAB, 0xF7, 0x15, 0x88, 0x09, 0xCF, 0x4

// LoRaWAN AppSKey, application session key
// This is the default Semtech key, which is used by the early prototype TTN
// network.
static const PROGMEM APPSKEY[16] = { 0x2B, 0x7E, 0x15, 0x16, 0x28, 0xAE, 0xD2, 0xA6, 0xAB, 0xF7, 0x15, 0x88, 0x09, 0xCF, 0x4

// LoRaWAN end-device address (DevAddr)
static const u4_t DEVADDR = 0x03FF0001; // <-- Change this address for every node!

// These callbacks are only used in over-the-air activation, so they are
// left empty here (we cannot leave them out completely unless
// DISABLE_JOIN is set in config.h, otherwise the linker will complain).
void os_getArtEui (ui_t* buf) { }
void os_getDevEui (ui_t* buf) { }
void os_getDevKey (ui_t* buf) { }

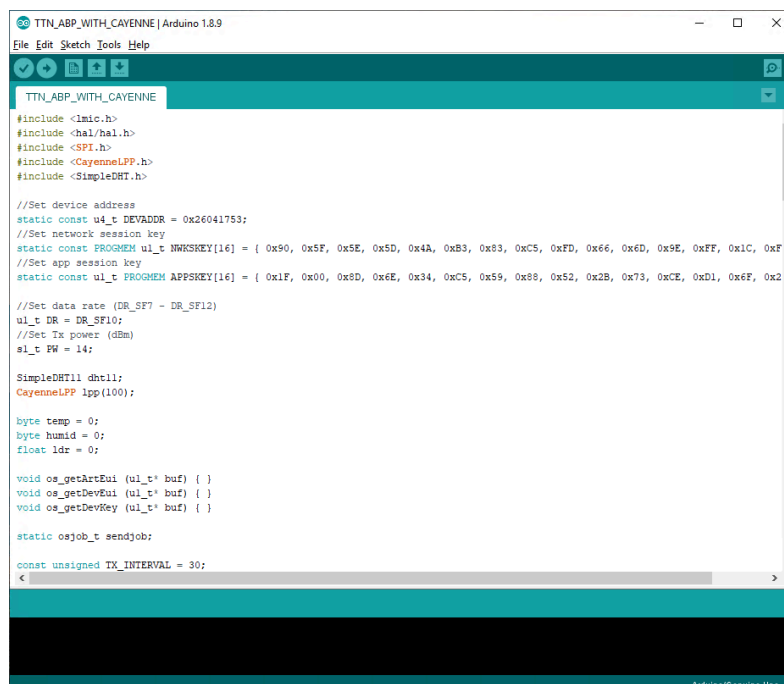
static uint8_t mydata[] = "Hello, world!";
static osjob_t sendjob;

// Schedule TX every this many seconds (might become longer due to duty
// cycle limitations).
const unsigned TX_INTERVAL = 60;

< >
```

รูปที่ 7 หน้าต่างโปรแกรม TTN_ABP

6. สำหรับโปรแกรม TTN_ABP_WITH_CAYENNE เมื่อเปิดแล้ว จะได้หน้าต่างดังรูปที่ 8 ซึ่งสามารถกำหนด Device address, Network session key, App session key, Spreading factor และกำลังส่งได้จากตัวแปร DEVADDR, NWKSKEY, APPSKEY, DR และ PW ตามลำดับ



```
TTN_ABP_WITH_CAYENNE | Arduino 1.8.9
File Edit Sketch Tools Help

TTN_ABP_WITH_CAYENNE

#include <lmic.h>
#include <hal/hal.h>
#include <SPI.h>
#include <CayenneLPP.h>
#include <SimpleDHT.h>

//Set device address
static const u4_t DEVADDR = 0x26041753;
//Set network session key
static const PROGMEM ui_t NWKSKEY[16] = { 0x90, 0x5F, 0x5E, 0x5D, 0x4A, 0xB3, 0x33, 0xC5, 0xFD, 0x66, 0x6D, 0x9E, 0xFF, 0x1C, 0xF
//Set app session key
static const PROGMEM APPSKEY[16] = { 0x1F, 0x00, 0x3D, 0x6E, 0x34, 0xC5, 0x59, 0x88, 0x52, 0x2B, 0x73, 0xCE, 0xD1, 0x6F, 0x2

//Set data rate (DR_SF7 - DR_SF12)
ui_t DR = DR_SF10;
//Set Tx power (dBm)
sl_t PW = 14;

SimpleDHT11 dht11;
CayenneLPP lpp(100);

byte temp = 0;
byte humid = 0;
float ldr = 0;

void os_getArtEui (ui_t* buf) { }
void os_getDevEui (ui_t* buf) { }
void os_getDevKey (ui_t* buf) { }

static osjob_t sendjob;

const unsigned TX_INTERVAL = 30;

< >
```

รูปที่ 8 หน้าต่างโปรแกรม TTN_ABP_WITH_CAYENNE