โครงงานสิ่งประดิษฐ์

โรงเรียนอิเล็กทรอนิกส์ กองวิทยาการ กรมอิเล็กทรอนิกส์ทหารเรือ

ชื่อโครงงาน(ภาษาไทย อุปกรณ์ตรวจวัดอุณหภูมิ 4 จุด

ชื่อโครงงาน(ภาษาอังกฤษ 4 – POSITION AMBIENT TEMPERATURE MEASUREMENT EQUIPMENT

ประเภทผลงาน สิ่งประดิษฐ์เพื่อพัฒนาต่อยอดนวัตกรรมและเทคโนโลยี

สถานะของผลงาน เป็นผลงานสิ่งประดิษฐ์ที่พัฒนาปรับปรุงแก้ไขใหม่

รายชื่อผู้ทำโครงงานสิ่งประดิษฐ์

๑. นรจ.ปณิธาน	นาคสวัสดิ์	ชั้น ๒ ห้อง จ	(หัวหน้ากลุ่ม
๒. นรจ.วันธงชัย	สุภวัฒนากร	ชั้น ๒ ห้อง จ	(รองหัวหน้ากลุ่ม
๓. นรจ.อรรถพล	ทรัพย์โภคา	ชั้น ๒ ห้อง จ	(สมาชิกกลุ่ม
๔. นรจ.อิทธิพล	เรืองเพ็ญ	ชั้น ๒ ห้อง จ	(สมาชิกกลุ่ม
๕. นรจ.อันวาร์	นฤนาทนามกร	ชั้น ๒ ห้อง จ	(สมาชิกกลุ่ม
๖. นรจ.ภูศิษฐ์	ชัยโพนงาม	ชั้น ๒ ห้อง จ	(สมาชิกกลุ่ม

ชื่อครูที่ปรึกษา

๑. น.ท. อุกฤษฏ์	อารมย์อุ่น
๒. พ.จ.ท. ชินเรษ	วิสียา
๓. จ.อ. ธิติรัตน์	เพชรพิทักษ์สิงห์

เกี่ยวกับโครงงาน

โครงงานสิ่งประดิษฐ์

ชื่อโครงงาน(ภาษาไทย **อุปกรณ์ตรวจวัดอุณหภูมิ 4 จุด**

ชื่อโครงงาน(ภาษาอังกฤษ 4 – POSITION AMBIENT TEMPERATURE MEASUREMENT EQUIPMENT

ผู้จัดทำ

1. นรจ.ปณิธาน นาคสวัสดิ์

2. นรจ.วันธงชัย สุภวัฒนากร

3. นรจ.อรรถพล ทรัพย์โภคา

4. นรจ.อิทธิพล เรืองเพ็ญ

5. นรจ.อันวาร์ นฤนาทนามกร

6. นรจ.ภูศิษฐ์ ชัยโพนงาม

ครูที่ปรึกษา

1. น.ท. อุกฤษฎ์ อารมย์อุ่น

2. พ.จ.ท. ชินเรษ วิสียา

3. จ.อ. ธิติรัตน์ เพชรพิทักษ์สิงห์

สถานศึกษา **โรงเรียนอิเล็กทรอนิกส์ กองวิทยาการ กรมอิเล็กทรอนิกส์ ทหารเรือ**

ปีการศึกษา 2/2562

สารบัญ

บทคัดย่อ ก

กิตติกรรมประกาศ ข

สารบัญ ค – จ

บทที่ **หน้า**

1 บทนำ

ความเป็นมาและความสำคัญของปัญหา 1

วัตถุประสงค์ของโครงงานสิ่งประดิษฐ์ 1

ขอบเขตของการศึกษา 1

นิยามศัพท์เฉพาะ 2

ประโยชน์ที่คาดว่าจะได้รับ 2

2 เอกสาร สิ่งประดิษฐ์และงานวิจัยที่เกี่ยวข้อง

โครงสร้างเซ็นเซอร์ตรวจจับอุณหภูมิและการติดตั้งเซ็นเซอร์ 3

บอร์ด ESP 8266 และการเขียนโปรแกรมลงบอร์ด 4

การเขียนโปรแกรม ESP8266 Arduino IDE 5 -17

โปรแกรม Sketchup 18-23

สิ่งประดิษฐ์ที่เกี่ยวข้องและงานวิจัยที่เกี่ยวข้อง IOT 24-25

บทที่ **หน้า**

3 วิธีดำเนินการโครงการสิ่งประดิษฐ์

การดำเนินงาน 26

FLOWCHART 27

วงจร Wiring Diagram 28

บล็อกไดอะแกรม 29

อุปกรณ์ 30 - 34

รายการอุปกรณ์ 35-38

แผนการดำเนินงาน 39-40

การทำงานของอุปกรณ์ตรวจวัดอุณหภูมิ 4 จุด 41

4 ผลการทดลอง

ผลการทดลอง 42-50

5 สรุปผล อภิปรายผลและข้อเสนอแนะ

สรุปผลและอภิปรายผล 51

วัสดุ อุปกรณ์ เครื่องมือหรือโปรแกรมที่ใช้ในการพัฒนา 51

สรุปผลการดำเนินงาน 52

ปัญหาและอุปสรรคในการจัดทำโครงการ 52

ข้อเสนอแนะ 52

บทที่ **หน้า**

ภาคผนวก

บทคัดย่อ

เนื่องจากการเรียนการสอนของหลักสูตร นรจ.อิเล็กทรอนิกส์ ชั้นปีที่ 2 ได้มีการให้ทำโครงงานสิ่งประดิษฐ์ก่อนจบการศึกษา โดยนำวิชาเรียนมาประยุกต์ให้เข้ากับโครงงานสิ่งประดิษฐ์ทางคณะผู้จัดทำได้เล็งเห็นความสำคัญของวิชา ป้องกันความเสียหาย จึงได้คิดค้นอุปกรณ์ที่เรียกว่า อุปกรณ์ตรวจวัดอุณหภูมิ 4 จุด เนื่องจากในปัจจุบันเทคโนโลยีอุปกรณ์อิเล็กทรอนิกส์พัฒนาหลากหลายและยังมีการปรับปรุงให้ทันสมัยมากขึ้น กับระบบต่างๆ ไม่ว่าจะเป็นระบบนิรภัย ระบบอำนวยความสะดวก จัดการสิ่งที่มนุษย์ทำได้ช้าหรือไม่สามรถทำได้ ทั้งนี้ทั้งนั้น พวกเราเล็งเห็นระบบนิรภัยมีความสำคัญ จึงได้คิดโครงงานนี้ขึ้นมา เพื่อช่วยเตือนป้องกันไฟไหม้ อย่างรวดเร็ว หรือยามวิกาล เวลาที่ไม่มีคนอยู่ในบริเวณไม่สามารถรับรู้เองได้

กิตติกรรมประกาศ

สิ่งประดิษฐ์เรื่อง อุปกรณ์ตรวจจับอุณหภูมิ 4 จุด ของนักเรียนจ่าอิเล็กทรอนิกส์ ชั้นปีที่ 2

ที่ได้รับการสนับสนุนจากโรงเรียนอิเล็กทรอนิกส์ กองวิทยาการกรมอิเล็กทรอนิกส์ทหารเรือ ที่ให้การสนับสนุนการทำโครงงานด้วยดีและให้ข้อเสนอแนะอันเป็นประโยชน์ ในการทำโครงงานครั้งนี้

คณะผู้จัดทำขอขอบพระคุณอาจารย์ปรึกษาโครงงานและอาจารย์ประจำวิชาที่ให้ คำแนะนำ ข้อเสนอแนะ พร้อมทั้งช่วยตรวจสอบ ปรับปรุงและแก้ไขรายงานโครงงานเล่มนี้อย่างดีมาตลอด จนกระทั่งทำ ให้การทำโครงงานครั้งนี้สำเร็จและสมบูรณ์มากยิ่งขึ้น

คณะผู้จัดทำยังได้รับความช่วยเหลือและความอนุเคราะห์จากคณะครูห้องทดลองที่ให้การสนับสนุนในการจัดหาและจัดซื้ออุปกรณ์เกี่ยวกับการทำโครงงานครั้งนี้

คณะผู้จัดทำขอขอบคุณในความอนุเคราะห์และความร่วมมือที่ดี

คณะผู้จัดทำ

กลุ่มที่ 15

อุปกรณ์ตรวจจับอุณหภูมิ 4 จุด

บทที่ 2

เอกสาร สิ่งประดิษฐ์และงานวิจัยที่เกี่ยวข้อง

การดำเนินโครงงานสิ่งประดิษฐ์ เรื่อง อุปกรณ์ตรวจจับอุณหภูมิ 4 จุด ผู้ดำเนินโครงงานสิ่งประดิษฐ์ได้ศึกษาเอกสาร สิ่งประดิษฐ์และงานวิจัยที่เกี่ยวข้อง โดยมีรายละเอียดดังนี้

1. โครงสร้างเซ็นเซอร์ตรวจจับอุณหภูมิและการติดตั้งเซ็นเซอร์
2. บอร์ด ESP 8266 และการเขียนโปรแกรมลงบอร์ด
3. โปรแกรม Sketchup
4. สิ่งประดิษฐ์ที่เกี่ยวข้องและงานวิจัยที่เกี่ยวข้อง

1. โครงสร้างเซ็นเซอร์ตรวจจับอุณหภูมิและการติดตั้งเซ็นเซอร์

MLX90614 เป็นเซ็นเซอร์วัดอุณหภูมิแบบไร้การสัมผัส โดยใช้หลักการแปลงแสงอินฟาเรดที่ส่งออกจากตัววัตถุซึ่งจะมีสีแตกต่างกัน ให้เป็นค่าอุณหภูมิ โมดูลวัดอุณหภูมิแบบอินฟาเรดไร้การสัมผัส ใช้ชิฟ MLX90614ESF สำหรับ Arduino ไฟเลี้ยง 3.3 V – 5 V เชื่อมต่อแบบ I2C ใช้สายเพียง 2 เส้นในการควบคุม สามารถวัดอุณหภูมิที่เป้าหมายแบบไร้การสัมผัสที่ - 70 ถึง 380 องศาเซลเซียส และยังสามารถวัดอุณหภูมิของสิ่งแวดล้อมได้ที่ -40 ถึง 125 องศาเซลเซียส ความละเอียดของอุณหภูมิที่วัดได้ 0.02 องศาเซลเซียส

1. ต่อวงจร

Node MCU ESP8266 -> GY-906 MLX90614 เซ็นเซอร์วัดอุณหภูมิแบบไร้สัมผัส

1. Vin(5V) -> VCC
2. GND -> GND
3. ขาD1 -> SCL
4. ขาD2 -> SDA

2. บอร์ด ESP 8266 และการเขียนโปรแกรมลงบอร์ด

ESP8266 เป็นชื่อเรียกของชิฟของโมดูล ESP8266 สำหรับติดต่อสื่อสารบนมาตรฐาน WIFI ทำงานที่แรงดันไฟฟ้า 3.3 – 5 V ทำงานใช้กระแสโดยเฉลี่ย 80mA รองรับคำสั่ง deep sleep ในการประหยัดพลังงาน ใช้

กระแสน้อยกว่า 10 ไมโครแอมป์ สามารถ wake up กลับมาส่งข้อมูลใช้เวลาน้อยกกว่า 2 มิลลิวินาที ภายในมี Low power MCU 32bit ทำให้เราเขียนโปรแกรมสั่งงานได้ มีวงจร analog digital converter ทำให้สามารถอ่านค่าจาก analog ได้ความละเอียด 10bit ทำงานได้ที่อุณหภูมิ -40 ถึง 125 องศาเซลเซียส รายละเอียดเพิ่มเติมจากผู้ผลิตอ้างอิงตามลิงค์นี้ ESP8266 Datasheet

เมื่อนำชิฟ ESP8266 มาผลิตเป็นโมดูลหลายรุ่น ก็จะขึ้นต้นด้วย ESP866 แล้วตามด้วยรุ่น เช่น ESP-01 , ESP-03 , ESP-07 , ESP-12E ESP8266 ติดต่อกับ WI-FI แบบ Serial สามารถเขียนโปรแกรมลงไปในชิฟ โดยใช้ Arduino IDE ได้ ทำให้การเขียนโปรแกรมและใช้งานเป็นเรื่องง่าย คล้ายกับการใช้ Arduino แน่นอนว่าสามารถติดต่ออุปกรณ์อื่น ๆ เซนเซอร์ ต่าง ๆ แบบสไตล์ Arduino ถ้ามีพื้นฐาน Arduino อยู่แล้ว ก็เข้าใจและใช้งานได้รวดเร็ว

โมดูล ESP8266 มีหลายรุ่น และมีรุ่นใหม่พัฒนาออกมาเรื่อย ๆ โดยโครงสร้างและขาที่ใช้งานก็จะมีลักษณะคล้ายกันคือ

- GPIO0 เป็นขาสำหรับเลือกโหมด โดยเมื่อต่อกับ GND จะเข้าโหมดโปรแกรม เมื่อต้องการให้ทำงานปกติก็ไม่ต้องต่อ
- GPIO15 เป็นขาที่ต้องต่อลง GND เพื่อให้โมดูลทำงาน

- CH_PD หรือ EN เป็นขาที่ต้องต่อไฟ VCC เพื่อ pull up สัญญาณ ให้โมดูลทำงาน โมดูลบางร่นไม่มีขา Reset มาให้ เมื่อต้องการรีเซต ให้ต่อขา CH_PD กับ GND
- Reset ต่อกับไฟ VCC เพื่อ pull up สัญญาณ โดยเมื่อต้องการรีเซต ให้ต่อกับไฟ GND
- VCC เป็นขาสำหรับจ่ายไฟเลี้ยง ใช้ไฟเลี้ยง 3.0-3.6V
- GND ต่อกับไฟ 0V
- GPIO เป็นขาดิจิตอล INPUT/OUTPUT ทำงานที่ไฟ 3.3V
- ADC เป็นขา Analog INPUT รับแรงดันสูงสุด 1V ความละเอียด 10bit หรือ 1024 ค่า

เวลาโปรแกรมเพียงมองหาขาเหล่านี้ แล้วต่อให้ครบเท่าที่มีขาให้ต่อ ก็สามารถโปรแกรม ESP8266 ได้ทุกรุ่น

การเขียนโปรแกรม ESP8266 Arduino IDE

สำหรับการต่อวงจรของ ESP8266 เพื่อเขียนโปรแกรมด้วย Arduino เขียนโคดเหมือนกันทุกรุ่น โดยแต่ละรุ่นจะมีขาไม่เท่ากัน ดังนั้นเราจึงต้องเปรียบเทียบขา GPIO ให้ถูกต้องในการสั่งงานก็ใช้ได้แล้ว การเขียนโปรแกรมอัพโหลดโคดลงบอร์ด ESP8266 เกือบทุกรุ่น จะผ่านทาง Serial ที่ขา rx, tx โดยใช้โมดูล USB TTL ซึ่งต้องเสียเวลาในการต่อวงจรเพื่ออัพโหลดโคด อีกทั้งโมดูล ESP8266 หลาย ๆ รุ่นมีการต่อขาที่เป็นแบบเซอร์เฟสเมาส์ ทำให้ไม่สะดวกกับการต่อทดลองบนบอร์ดทดลอง

ดังนั้นจึงมีการรวม โมดูล USB TTL และต่อวงจรขยายขา ESP8266 ให้เป็นขาระยะห่างขนาด 2.54mm ซึ่งสามารถเสียบลงบอร์ดทดลองได้พอดี กลายเป็น บอร์ด ESP8266
โดยหนึ่งในบอร์ดที่นิยมใช้งานคือ Node MCU ซึ่งใช้ โมดูล ESP8266 ESP-12E

การเขียน โปแกรมลง NodeMCU ESP8266

1. ดาวน์โหลด Library

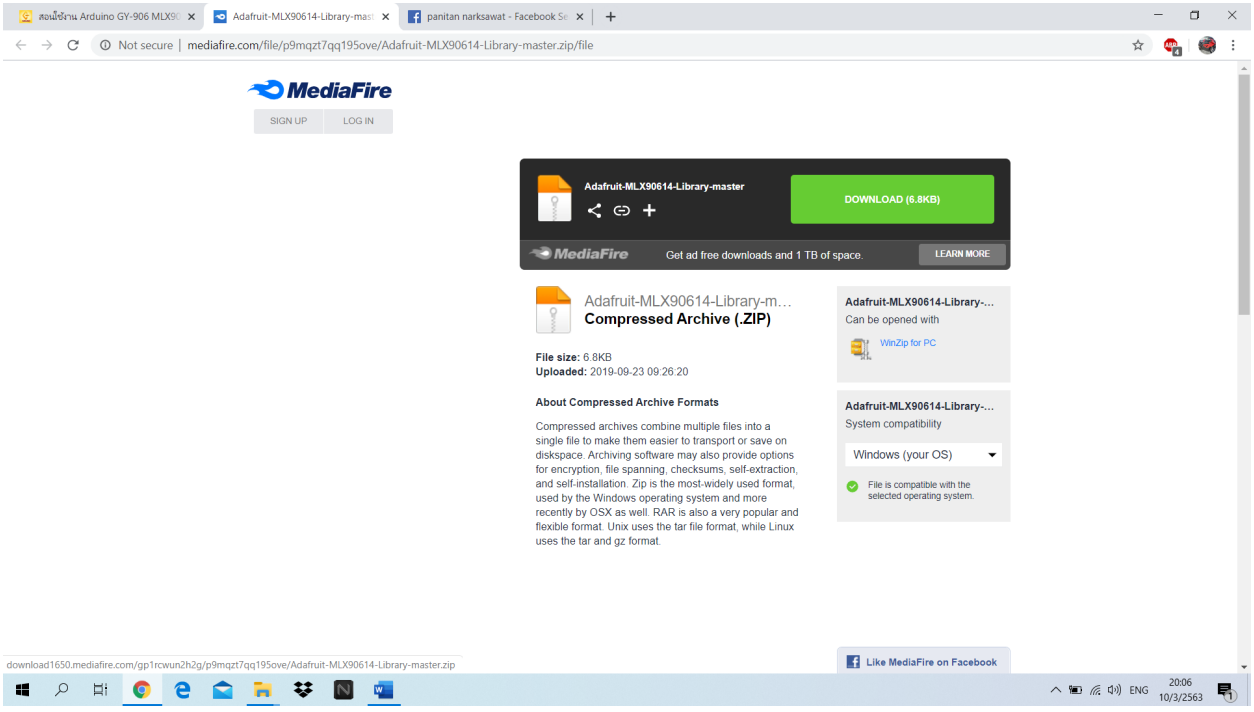
1.1 วิธีการโหลด Library

1.1.1. กดไปที่ Library

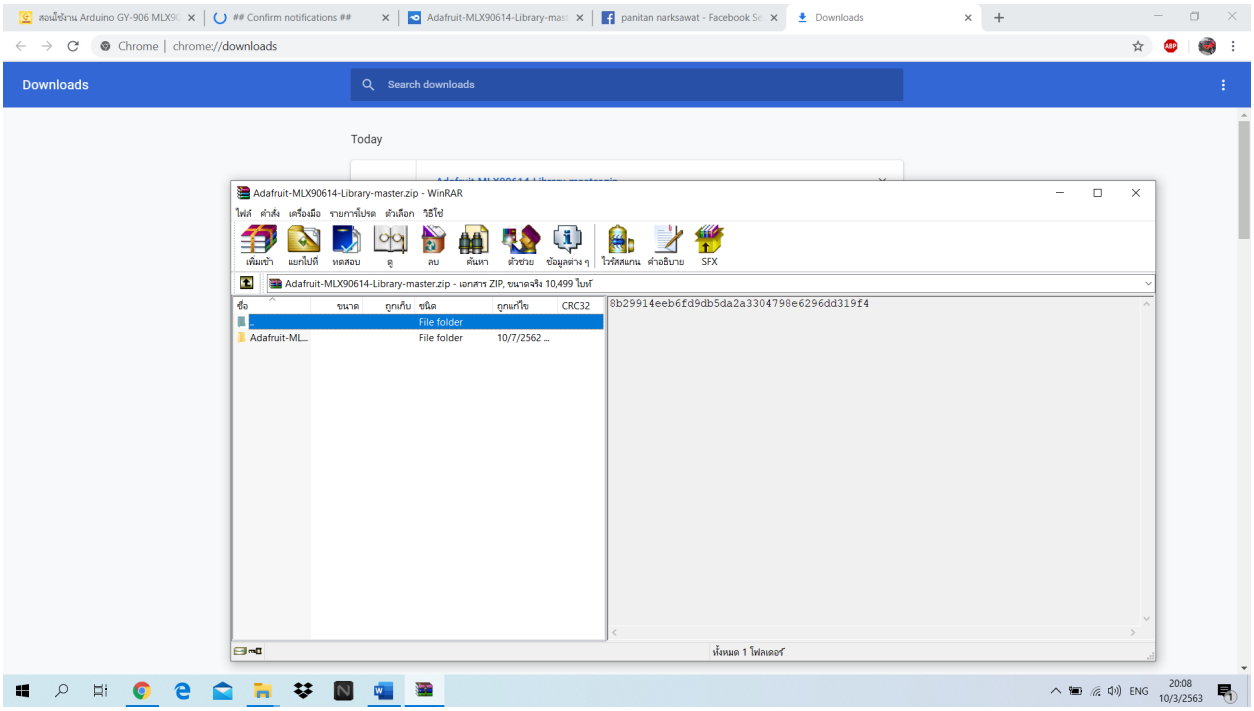
ดาวน์โหลด Library GY-906 MLX90614 เซ็นเซอร์วัดอุณหภูมิแบบไร้สัมผัส

- http://www.mediafire.com/file/p9mqzt7qq195ove/Adafruit-MLX90614-Library-master.zip/file

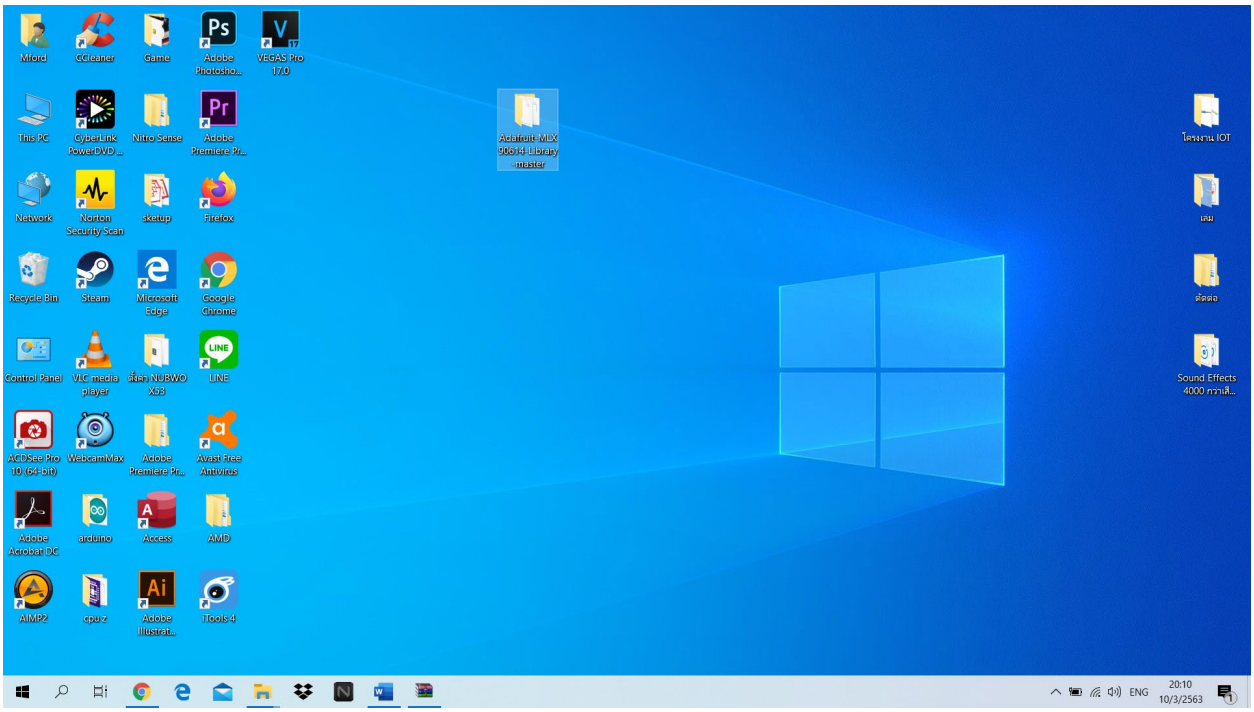
1.1.2. กดไปที่ Download



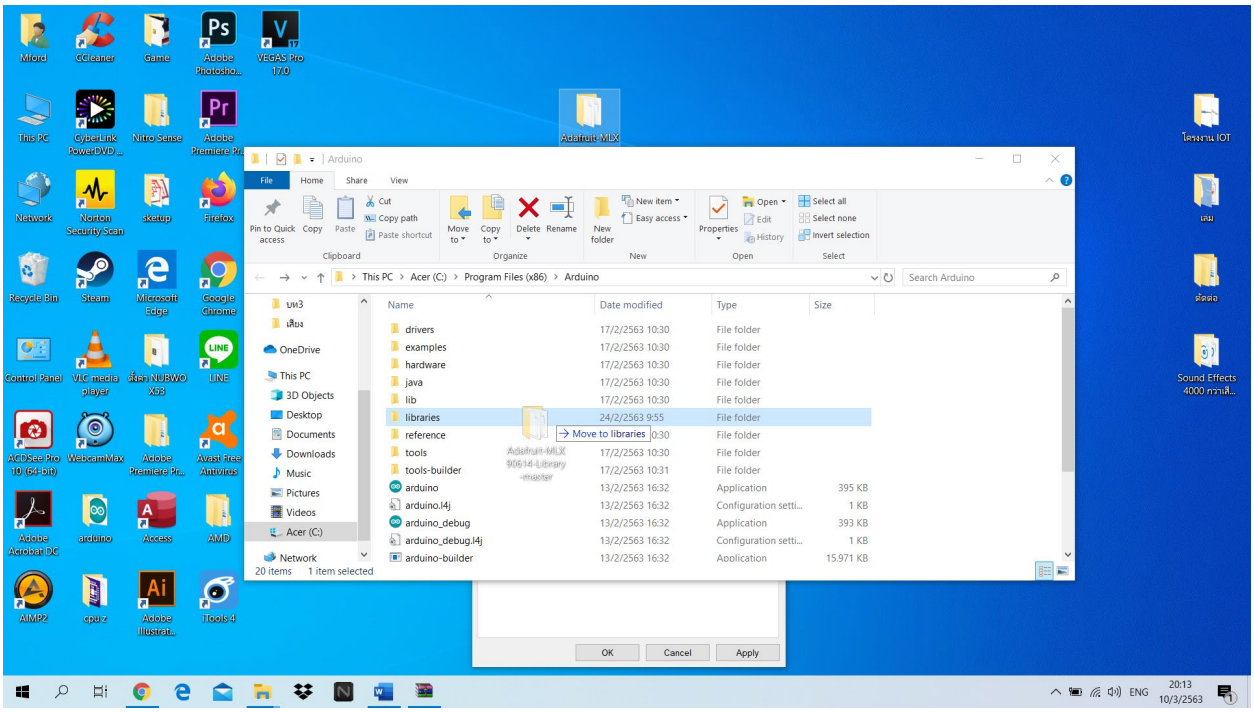
1.1.3. เมื่อโหลดเสร็จแล้วให้แตกไฟล์



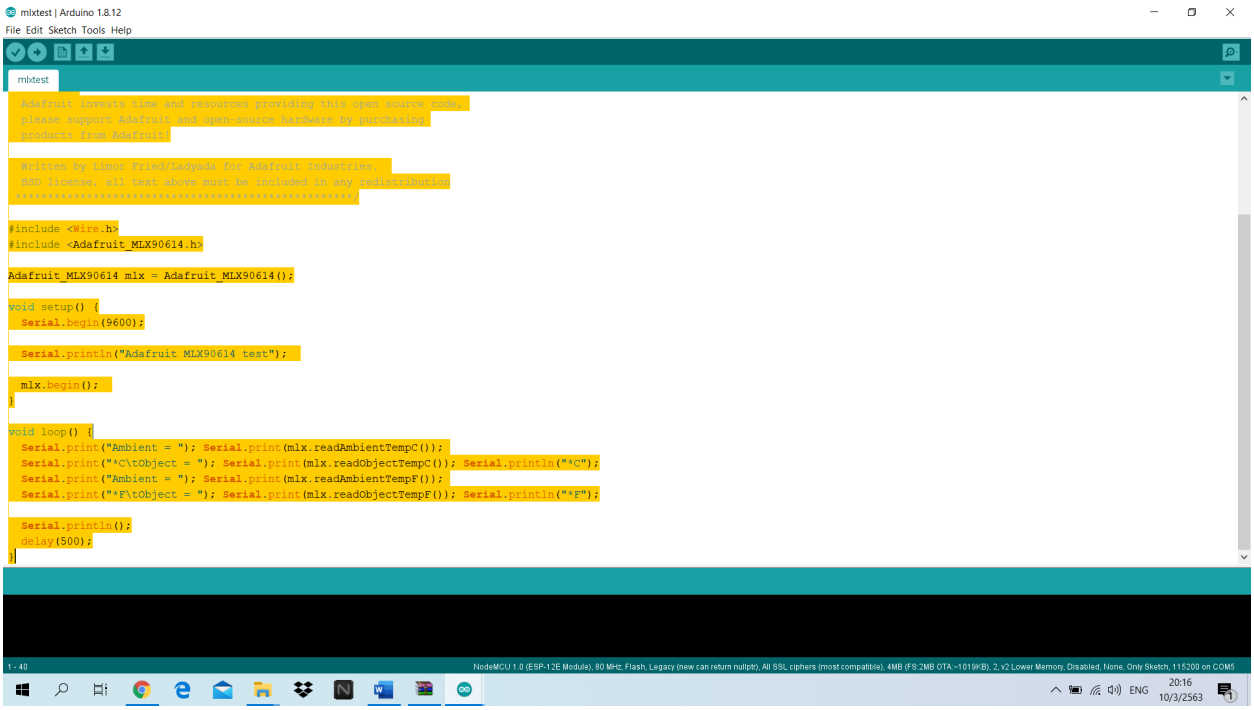
1.1.4. ให้เข้าไปใน libraries ของ Arduino



1.1.5. ลากไฟล์ที่แตกมาลงใน libraries



Code ที่เราใช้



1.1.6. Code Client ที่เราใช้

```
#include <ESP8266WiFi.h>
#include <WiFiUDP.h>
#include <TridentTD_LineNotify.h>
//#define SSID "Wifi0_2.4G" //SSID Wifi
//#define PASSWORD "ball3123" //Pass Wifi
#define LINE_TOKEN "gLkCAAPozce1OhaE2ATNpSJ7ryXU8vj7ttj9ZtdS9NK" //Token ที่ได้จากการสมัคร ID Line
#include <Wire.h>
#include <Adafruit_MLX90614.h>
#define IR1 0x5A
#define IR2 0x5B
#define IR3 0x5C
#define IR4 0x5D
#define udp_port 40000
Adafruit_MLX90614 mlx;
int buzzer_module = 16;
int ledPin1 = 00;
int ledPin2 = 02;
unsigned int UDPPort = 40000; // local port to listen on
char ssid[] = "iPhone";
char pass[] = "ball3123";
char remoteIp[] = "172.20.10.3";
char packetBuffer[255]; //buffer to hold incoming packet
char replyBuffer[] = "acknowledged"; // a string to send back

WiFiUDP Udp;

void setup() {
Serial.begin(115200);
WiFi.begin(ssid,pass);

Serial.println();
Serial.print("Wait for WiFi");

while (WiFi.status() != WL_CONNECTED) {
delay(500);
Serial.print(".");
}
Serial.println("");
Serial.println("WiFi connected");
Serial.println("IP address: " + WiFi.localIP().toString());

Udp.begin(UDPPort);

Udp.beginPacket("172.20.10.3", UDPPort);//send ip to server
char ipBuffer[255];
WiFi.localIP().toString().toCharArray(ipBuffer, 255);
```

```
Udp.write(ipBuffer);
Udp.endPacket();
Serial.println("Sent ip adress to server");
//led
{pinMode(ledPin1, OUTPUT);
pinMode(ledPin2, OUTPUT);}
//buzzer
{pinMode(buzzer_module, OUTPUT);}

Serial.begin(115200);
//sensor
mlx.begin();

Serial.println("Adafruit MLX90614 test");

//line
Serial.println(LINE.getVersion());
WiFi.begin(ssid, pass);
Serial.printf("WiFi connecting to %s\n", ssid);
while (WiFi.status() != WL_CONNECTED) {
Serial.print(".");
delay(1000);
}
Serial.printf("\nWiFi connected\nIP : ");
Serial.println(WiFi.localIP());

// กำหนด Line Token
LINE.setToken(LINE_TOKEN);
LINE.notify("Temp");
}

void loop()
{



mlx.AddrSet(IR1);

Serial.print("IR1: ");
Serial.print("Ambient = ");
Serial.print(mlx.readAmbientTempC());
Serial.print("*C\tObject = ");
Serial.print(mlx.readObjectTempC());
Serial.println("*C");
Serial.print("Ambient = ");
Serial.print(mlx.readAmbientTempF());
Serial.print("*F\tObject = ");
Serial.print(mlx.readObjectTempF());
```

```
Serial.println("*F");
mlx.temp1 = mlx.readAmbientTempC();
Serial.println();
delay(250);

float AmbientTempC_A = mlx.readAmbientTempC();
float AmbientTempF_A = mlx.readAmbientTempF();
float ObjectTempC_A = mlx.readObjectTempC ();
float ObjectTempF_A = mlx.readObjectTempF ();

mlx.AddrSet(IR2);

Serial.print("IR2: ");
Serial.print("Ambient = ");
Serial.print(mlx.readAmbientTempC());
Serial.print("*C\tObject = ");
Serial.print(mlx.readObjectTempC());
Serial.println("*C");
Serial.print("Ambient = ");
Serial.print(mlx.readAmbientTempF());
Serial.print("*F\tObject = ");
Serial.print(mlx.readObjectTempF());
Serial.println("*F");
mlx.temp2 = mlx.readAmbientTempC();
Serial.println();
delay(250);

float AmbientTempC_B = mlx.readAmbientTempC();
float AmbientTempF_B = mlx.readAmbientTempF();
float ObjectTempC_B = mlx.readObjectTempC ();
float ObjectTempF_B = mlx.readObjectTempF ();

mlx.AddrSet(IR3);

Serial.print("IR3: ");
Serial.print("Ambient = ");
Serial.print(mlx.readAmbientTempC());
Serial.print("*C\tObject = ");
Serial.print(mlx.readObjectTempC());
Serial.println("*C");
Serial.print("Ambient = ");
Serial.print(mlx.readAmbientTempF());
Serial.print("*F\tObject = ");
Serial.print(mlx.readObjectTempF());
Serial.println("*F");
mlx.temp3 = mlx.readAmbientTempC();
Serial.println();
delay(250);
```

```
float AmbientTempC_C = mlx.readAmbientTempC();
float AmbientTempF_C = mlx.readAmbientTempF();
float ObjectTempC_C = mlx.readObjectTempC ();
float ObjectTempF_C = mlx.readObjectTempF ();

mlx.AddrSet(IR4);

Serial.print("IR4: ");
Serial.print("Ambient = ");
Serial.print(mlx.readAmbientTempC());
Serial.print("*C\tObject = ");
Serial.print(mlx.readObjectTempC());
Serial.println("*C");
Serial.print("Ambient = ");
Serial.print(mlx.readAmbientTempF());
Serial.print("*F\tObject = ");
Serial.print(mlx.readObjectTempF());
Serial.println("*F");
mlx.temp4 = mlx.readAmbientTempC();
Serial.println();
delay(250);

float AmbientTempC_D = mlx.readAmbientTempC();
float AmbientTempF_D = mlx.readAmbientTempF();
float ObjectTempC_D = mlx.readObjectTempC ();
float ObjectTempF_D = mlx.readObjectTempF ();



//แสดงค่าเฉลี่ยอุณภูมิห้อง
//Serial.print("\n----------Average Temp:");
// Serial.print(mlx.avgF=(mlx.temp1+mlx.temp2+mlx.temp3+mlx.temp4)/4);
//Serial.print("----------\n");

float (mlx.avgF=(mlx.temp1+mlx.temp2+mlx.temp3+mlx.temp4)/4);
if ((AmbientTempC_A > 40 )||(AmbientTempC_B > 40)||(AmbientTempC_C >
40)||(AmbientTempC_D > 40))
{
//buzerr ทำงาน
digitalWrite(buzzer_module, HIGH);
digitalWrite(ledPin2, LOW);
//led แจ้งเตือน

digitalWrite(ledPin1, HIGH); // set the LED on
delay(200); // wait for a second
digitalWrite(ledPin1, LOW); // set the LED off
delay(200);
```

```
String LineText6;
//ถ้า A เกินส่งทุกตัวนี้
if (AmbientTempC_A > 40 ){
String string10 = "อุณหภูมิตัวที่ 1 เกินกำหนด !! ";
String string11 = " °C";
LineText6 = string10 + AmbientTempC_A + string11;
Serial.print("Line ");
Serial.println(LineText6);
LINE.notify(LineText6);}

digitalWrite(ledPin1, HIGH); // set the LED on
delay(200); // wait for a second
digitalWrite(ledPin1, LOW); // set the LED off
delay(200);
String string12 = "อุณหภูมิห้องตัวที่ 1 ";
String string13 = " °C";
LineText6 = string12 + AmbientTempC_A + string13;
Serial.print("Line ");
Serial.println(LineText6);
LINE.notify(LineText6);
digitalWrite(ledPin1, HIGH); // set the LED on
delay(200); // wait for a second
digitalWrite(ledPin1, LOW); // set the LED off
delay(200);
String LineText7;

if (AmbientTempC_B > 40){
String string14 = "อุณหภูมิตัวที่ 2 เกินกำหนด !!! ";
String string15 = " °C";
LineText7 = string14 + AmbientTempC_B + string15;
Serial.print("Line ");
Serial.println(LineText7);
LINE.notify(LineText7);}

String string16 = "อุณหภูมิห้องตัวที่ 2 ";
String string17 = " °C";
LineText7 = string16 + AmbientTempC_B + string17;
Serial.print("Line ");
Serial.println(LineText7);
LINE.notify(LineText7);

digitalWrite(ledPin1, HIGH); // set the LED on
delay(200); // wait for a second
digitalWrite(ledPin1, LOW); // set the LED off
delay(200);
String LineText8;

if (AmbientTempC_C > 40){
```

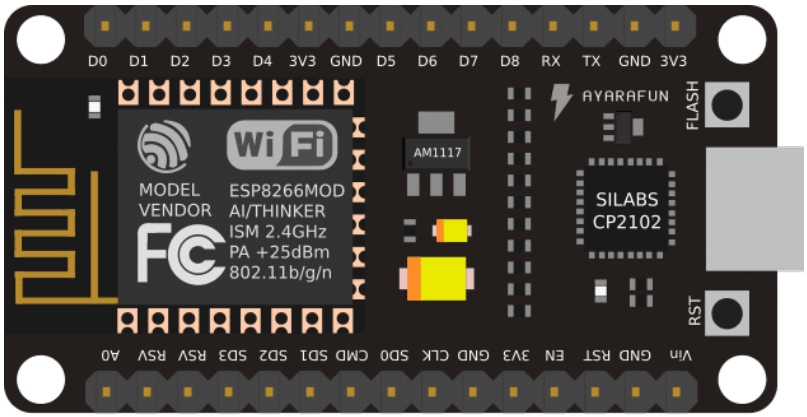
```
String string18 = "อุณหภูมิตัวที่ 3 เกินกำหนด !!! ";
String string19 = " °C";
LineText8 = string18 + AmbientTempC_C + string19;
Serial.print("Line ");
Serial.println(LineText8);
LINE.notify(LineText8);}
digitalWrite(ledPin1, HIGH); // set the LED on
delay(200); // wait for a second
digitalWrite(ledPin1, LOW); // set the LED off
delay(200);
String string20 = "อุณหภูมิห้องตัวที่ 3 ";
String string21 = " °C";
LineText8 = string20 + AmbientTempC_C + string21;
Serial.print("Line ");
Serial.println(LineText8);
LINE.notify(LineText8);
digitalWrite(ledPin1, HIGH); // set the LED on
delay(200); // wait for a second
digitalWrite(ledPin1, LOW); // set the LED off
delay(200);
String LineText9;

if (AmbientTempC_D > 40){
String string22 = "อุณหภูมิตัวที่ 4 เกินกำหนด !!! ";
String string23 = " °C";
LineText9 = string22 + AmbientTempC_D + string23;
Serial.print("Line ");
Serial.println(LineText9);
LINE.notify(LineText9);}
digitalWrite(ledPin1, HIGH); // set the LED on
delay(200); // wait for a second
digitalWrite(ledPin1, LOW); // set the LED off
delay(200);
String string24 = "อุณหภูมิห้องตัวที่ 4 ";
String string25 = " °C";
LineText9 = string24 + AmbientTempC_D + string25;
Serial.print("Line ");
Serial.println(LineText9);
LINE.notify(LineText9);
digitalWrite(ledPin1, HIGH); // set the LED on
delay(200); // wait for a second
digitalWrite(ledPin1, LOW); // set the LED off
delay(200);
String LineText5;

String string28 = "Ambient Avg Temp = ";

LineText5 = string28 + (mlx.avgF=(mlx.temp1+mlx.temp2+mlx.temp3+mlx.temp4)/4) +
```

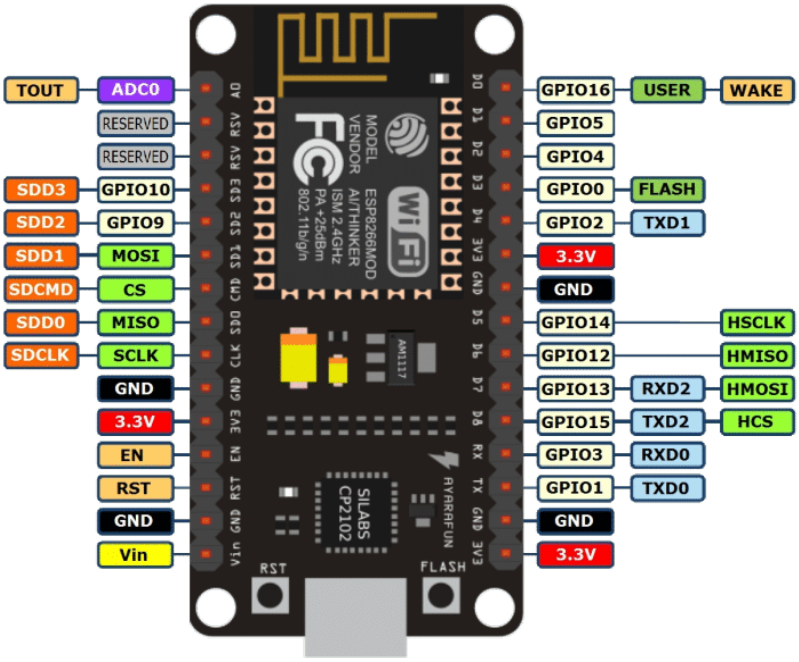
```
string25;
LINE.notify(LineText5);

digitalWrite(ledPin1, HIGH); // set the LED on
delay(200); // wait for a second
digitalWrite(ledPin1, LOW); // set the LED off
delay(200);
delay(1000);}

else
{
String LineText;
String string0 = "°C ] ";
String string1 = "°F/ ";
String string2 = " [AmbientTemp A = ";
String string3 = " [ObjectTemp A = ";
LineText = string2 + AmbientTempF_A + string1 + AmbientTempC_A + string0
+ string3 + ObjectTempF_A + string1 + ObjectTempC_A + string0 ;
Serial.print("Line ");
Serial.println(LineText);
LINE.notify(LineText);


String LineText1;
String string4 = " [AmbientTemp B = ";
String string5 = " [ObjectTemp B = ";
LineText1 = string4 + AmbientTempF_B + string1 + AmbientTempC_B + string0
+ string5 + ObjectTempF_B + string1 + ObjectTempC_B + string0 ;
Serial.print("Line ");
Serial.println(LineText1);
LINE.notify(LineText1);



String LineText2;
String string6 = " [AmbientTemp C = ";
String string7 = " [ObjectTemp C = ";
LineText2 = string6 + AmbientTempF_C + string1 + AmbientTempC_C + string0
+ string7 + ObjectTempF_C + string1 + ObjectTempC_B + string0 ;
Serial.print("Line ");
Serial.println(LineText2);
LINE.notify(LineText2);



String LineText3;
String string8 = " [AmbientTemp D = ";
String string9 = " [ObjectTemp D = ";
```

```
LineText3 = string8 + AmbientTempF_D + string1 + AmbientTempC_D + string0
+ string9 + ObjectTempF_D + string1 + ObjectTempC_D + string0 ;
Serial.print("Line ");
Serial.println(LineText3);
LINE.notify(LineText3);

String LineText4;

String string27 = "Ambient Avg Temp = ";

LineText4 = string27 + (mlx.avgF=(mlx.temp1+mlx.temp2+mlx.temp3+mlx.temp4)/4) +
string0;
LINE.notify(LineText4);


digitalWrite(buzzer_module, LOW);
digitalWrite(ledPin2, HIGH);
digitalWrite(ledPin1, LOW);
delay (60000); }


// if there's data available, read a packet
Udp.beginPacket(remoteIp, UDPPort);
//Udp.write("$TEMP,Station1,<temp1>,<temp2>,<temp3>,<temp4><avg>*");
Udp.write("[sensor 1 = ");
Udp.println(AmbientTempC_A);
Udp.write("*C");
Udp.write("]");
Udp.println();
Udp.write("[sensor 2 = ");
Udp.println(AmbientTempC_B);
Udp.write("*C");
Udp.write("]");
Udp.println();
Udp.write("[sensor 3 = ");
Udp.println(AmbientTempC_C);
Udp.write("*C");
Udp.write("]");
Udp.println();
Udp.write("[sensor 4 = ");
Udp.println(AmbientTempC_D);
Udp.write("*C");
Udp.write("]");
Udp.println();
Udp.write("[Avg = ");
Udp.println(mlx.avgF=(mlx.temp1+mlx.temp2+mlx.temp3+mlx.temp4)/4);
Udp.write("*C");
Udp.write("]");
```

```
Udp.println();
Udp.endPacket();
delay(5000);

}
```

บอร์ด ESP8266 NodeMCU V2





Node MCU V2 เป็น ESP8266-12E รวมกับ USB TTL ที่ใช้ชิฟ CP2102 และขยายขาให้สามารถต่อทดลองได้ง่ายขึ้น มีปุ่ม reset และ flash สำหรับใช้โปรแกรม โดยใช้ Arduino IDE หรือโปรแกรมอื่น ๆ ได้อย่างสะดวก

Node MCU V3 เป็นบอร์ดทีคล้ายกับ Node MCU V2 ที่ต่างกันคือ Node MCU V3 จะมีขนาดกว้างกว่า และใช้ชิฟ USB TTL เป็น CH340 ซึ่งการต่อขาใช้งานและโคดโปรแกรมเหมือนกันทุกประการ

3. โปรแกรม Sketchup

SketchUp คืออะไร? ความเป็นมาของ SketchUp

SketchUp Make เป็นโปรแกรมออกแบบที่มีความสามารถในการเปลี่ยนภาพวาดโครงร่างให้กลายเป็นภาพงานจำลอง 3 มิติ เป็นโปรแกรมขนาดเล็ก จึงทำให้มีการประมวลผลออกมาอย่างรวดเร็วถ้าเป็นรุ่น SketchUp Pro จะต้องเสียเงินซื้อ แต่ถ้าเป็นรุ่นทั่วไปจะมีให้โหลดใช้ฟรีค่ะ และเพราะความฟรี บวกกับความง่ายในการใช้งานของมันนี่เอง จึงทำให้นักออกแบบคนไทยส่วนมากนิยมหันมาใช้เจ้าโปรแกรมตัวนี้กันมากขึ้น

3.1. ความเป็นมาของโปรแกรม SketchUp

เริ่มแรกโปรแกรม SketchUp ถูกพัฒนาขึ้นโดยบริษัท @Last Software ในปี ค.ศ. 2000 ด้วยแนวคิด “3D for everyone” (3D สำหรับทุกคน จากนั้นทาง Google ก็เกิดสนใจเจ้าโปรแกรมตัวนี้เข้าเพราะมันมีคุณสมบัติหลายอย่างตรงกับที่ต้องการ เช่น ใช้งานง่าย ใช้ทรัพยากรเครื่องต่ำ กินเนื้อที่เครื่องน้อย ประมวลผลได้อย่างรวดเร็ว ด้วยเหตุนี้โปรแกรม SketchUp จึงเหมาะอย่างยิ่งที่จะนำมาใช้เป็นเครื่องมือสำหรับให้คนทั่วไปใช้สร้างแผนที่ 3D บน Google Maps และ Google Earth ซึ่ง 2 อย่างนี้ถือว่าเป็นเรื่องใหม่เอามาก ๆ ในสมัยนั้นในปี 2006 ทาง Google จึงได้ซื้อโปรแกรม SketchUp จากบริษัท @Last Software มาและเปลี่ยนชื่อเป็น Google SketchUp เพื่อนำมันมาพัฒนาต่อและ ให้มันสามารถใช้สร้างแผนที่ 3D ให้ได้ตามที่ตั้งใจไว้แม้ว่า Google SketchUp จะถูกพัฒนาจนมาถึงเวอร์ชั่น 8 และทาง Google ยังคงนำมันมาใช้สร้างแผนที่ 3D อยู่ แต่ทว่าหลังจากนั้น Google ก็ไม่มีนโนบายชัดเจนที่จะนำมันมาพัฒนาต่อแบบจริงจัง ดังนั้นทีมงานส่วนหนึ่งของ Google SketchUp ซึ่งบางคนก็เป็นหนึ่งในทีมพัฒนามาตั้งแต่ยุคเริ่มของบริษัท @Last Software จึงได้ย้ายไปอยู่กับบริษัท Trimbleโดยในปี 2012 บริษัท Trimble ได้ซื้อโปรแกรม Google SketchUp มาและให้สัญญาว่าจะนำมันมาพัฒนาต่ออย่างจริงจังปัจจุบัน Trimble ได้นำจุดแข็งของตนนั่นคือ ความเชี่ยวชาญด้านวิศวกรรม มาผนวกกับจุดแข็งของทีมงาน SketchUp เก่าที่มีความเชี่ยวชาญด้านโปรแกรมกราฟฟิก เมื่อจับทั้งสองฝ่ายมาร่วมมือกันจึงทำให้ SketchUp ครอบคลุมทุกสิ่งที่นักออกแบบ และนักวิศวกรกราฟิกต้องการมากขึ้นเรื่อย ๆ

3.2. คุณสมบัติของโปรแกรม SketchUp Make

– มีเครื่องมือหลายตัวที่ช่วยอำนวยความสะดวกในการออกแบบให้ง่ายขึ้น

– มีการออกแบบเมนูเครื่องมือที่ในการวาดแบบ ให้คล้ายกับของโปรแกรม Photoshop หรือ Paint ใน Windows เพื่อย่นระยะในการทำความเข้าใจ และง่ายในการเรียนรู้

– สามารถหมุนดูวัตถุ 3 มิติ หรือแบบ 2 มิติได้ 360 องศา อีกทั้งสามารถซูมเข้า - ออกได้ตามต้องการ

– มี VDO สอนการใช้งานอย่างละเอียด ทั้งจากในโปรแกรม หรือจาก YouTube พร้อมวิธีให้ลองทำตาม

– สามารถใส่เงาของวัตถุที่ออกแบบได้อย่างอัตโนมัติ

– สามารถย้อนกลับการทำงานได้หลายขั้นตอน

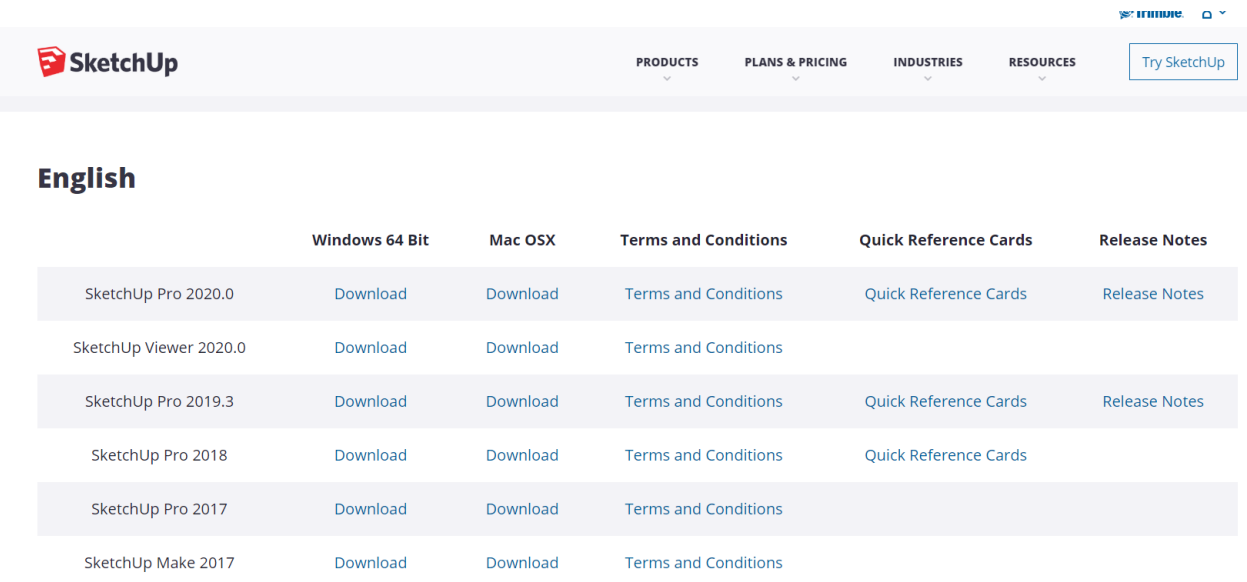
– สามารถเลือกสีและปรับแต่งชิ้นงานได้อย่างอิสระ

– โหลดใช้ได้ฟรี ไม่มีค่าใช้จ่ายแอบแฝง

– รองรับการทำงานทั้งบนระบบปฏิบัติการ Windows และ Mac OS X

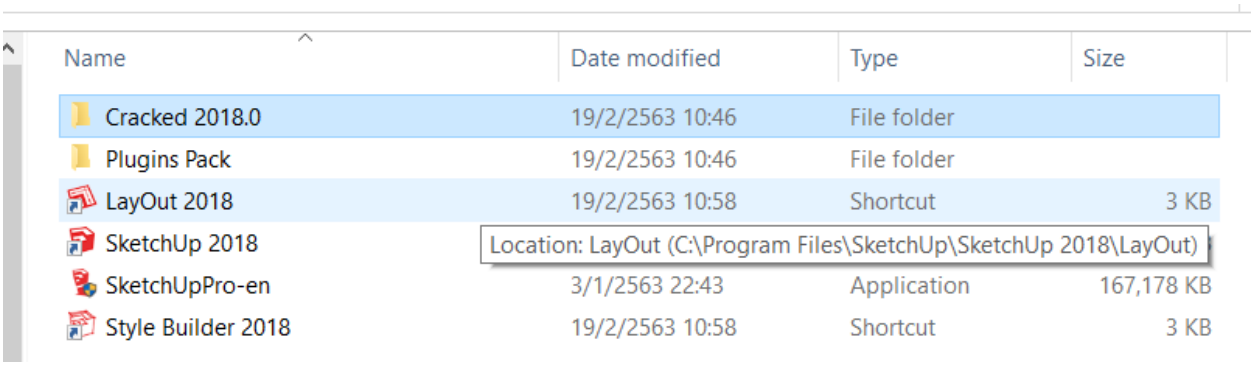
3.3. วิธีการโหลด

3.3.1. กด DOWNLOAD ให้ตรงกับสเปคคอมของตัวเอง



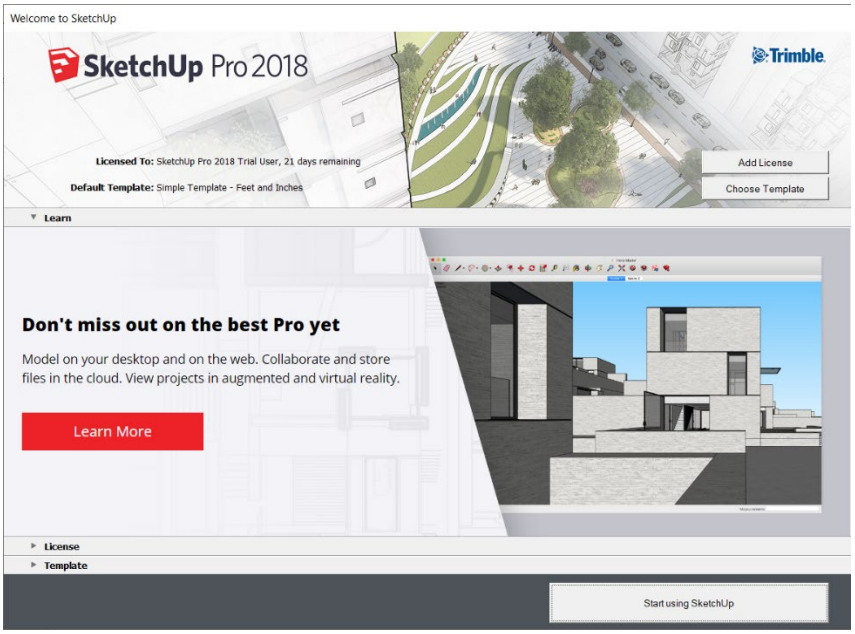
	Windows 64 Bit	Mac OSX	Terms and Conditions	Quick Reference Cards	Release Notes
SketchUp Pro 2020.0	Download	Download	Terms and Conditions	Quick Reference Cards	Release Notes
SketchUp Viewer 2020.0	Download	Download	Terms and Conditions		
SketchUp Pro 2019.3	Download	Download	Terms and Conditions	Quick Reference Cards	Release Notes
SketchUp Pro 2018	Download	Download	Terms and Conditions	Quick Reference Cards	
SketchUp Pro 2017	Download	Download	Terms and Conditions		
SketchUp Make 2017	Download	Download	Terms and Conditions		

3.3.2. รอโหลดซักครู่ จะได้โปรแกรมมาแล้วก็ลงโปรแกรมตามลำดับ

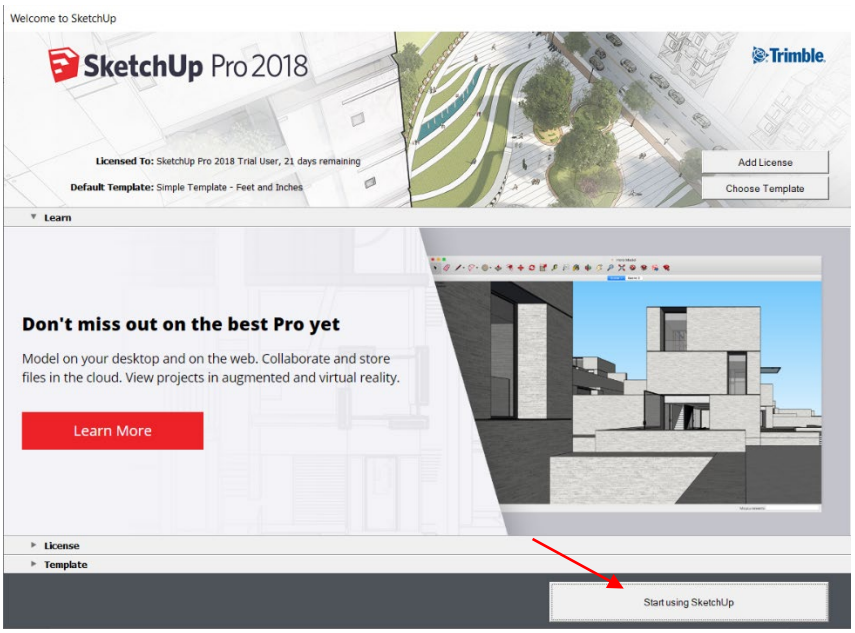


3.4. วิธีการใช้งานโปรแกรม SketchUp

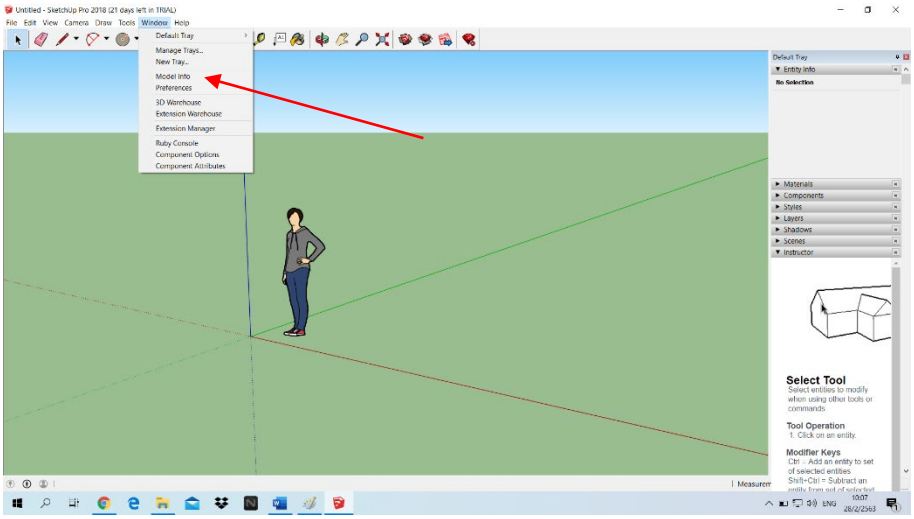
3.4.1. เปิดโปรแกรม SketchUp



3.4.2 กดที่ Start using SketchUp

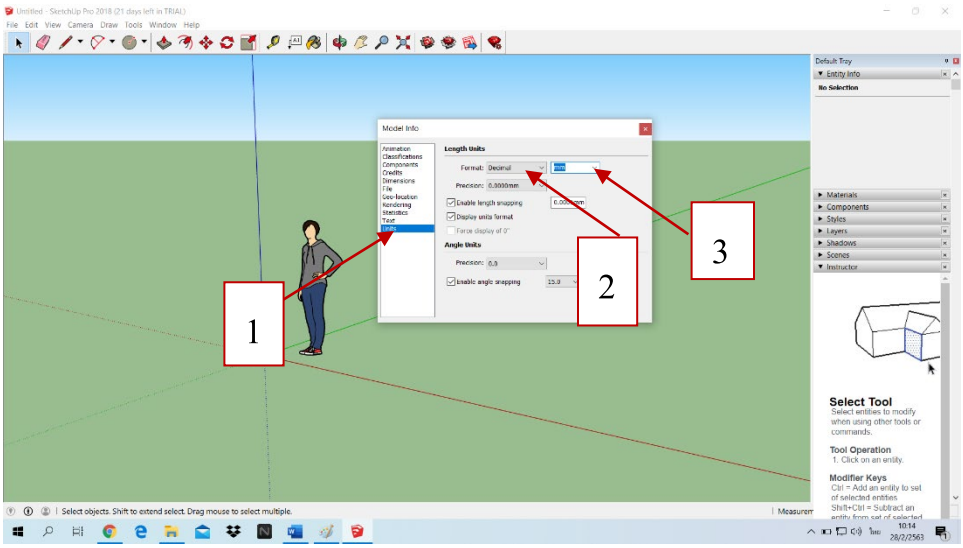


3.4.3. เลือกเมนูไปที่ Window เข้าไปที่ Model Info เพื่อตั้งค่าของชิ้นงาน

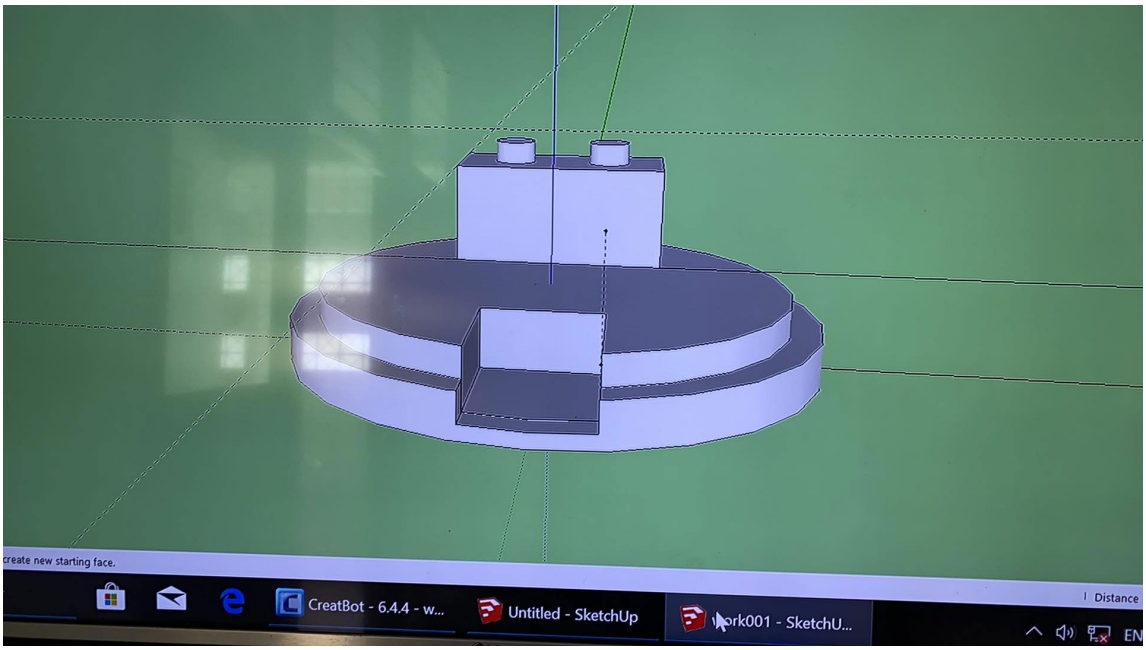


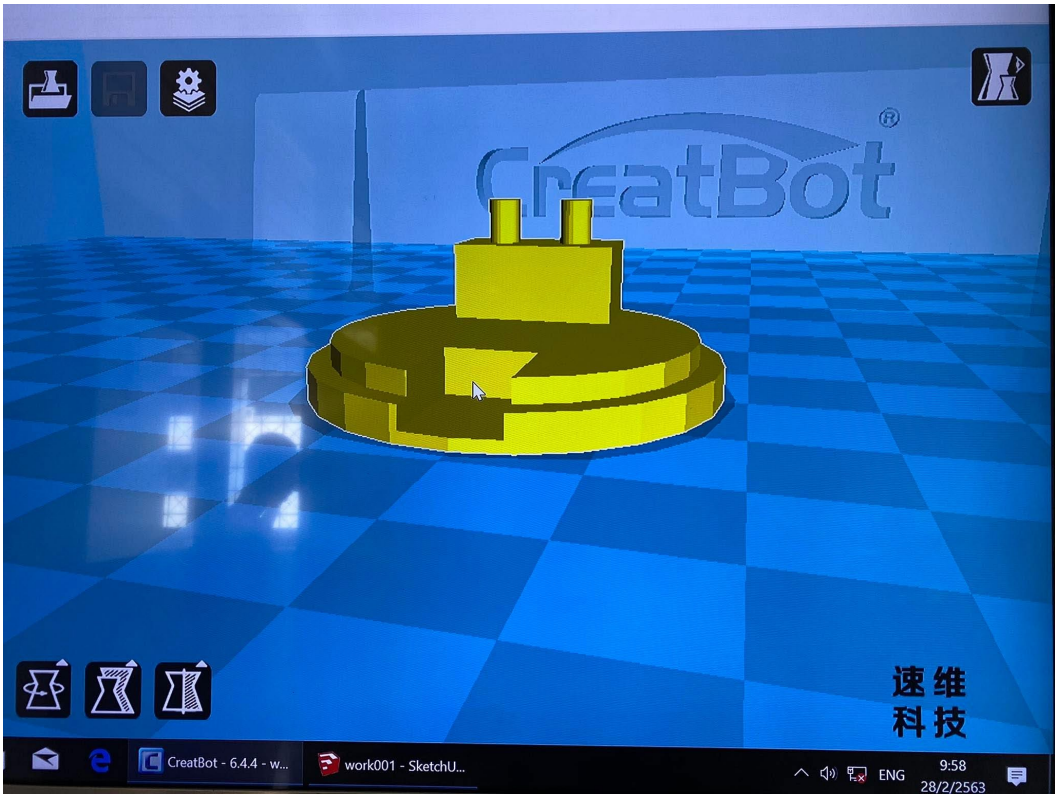
3.4.4. เมื่อหน้าต่าง Model Info ขึ้นแล้ว ให้เลือก ตามขั้นตอน

1. เลือกที่ Units
2. เลือก Format เปลี่ยนเป็น Decimal
3. เปลี่ยนจาก Inches เป็น mm
4. เสร็จสิ้นการตั้งค่าชิ้นงาน



3.5.5. ตัวอย่างการออกแบบเขียนโปรแกรม SketchUp



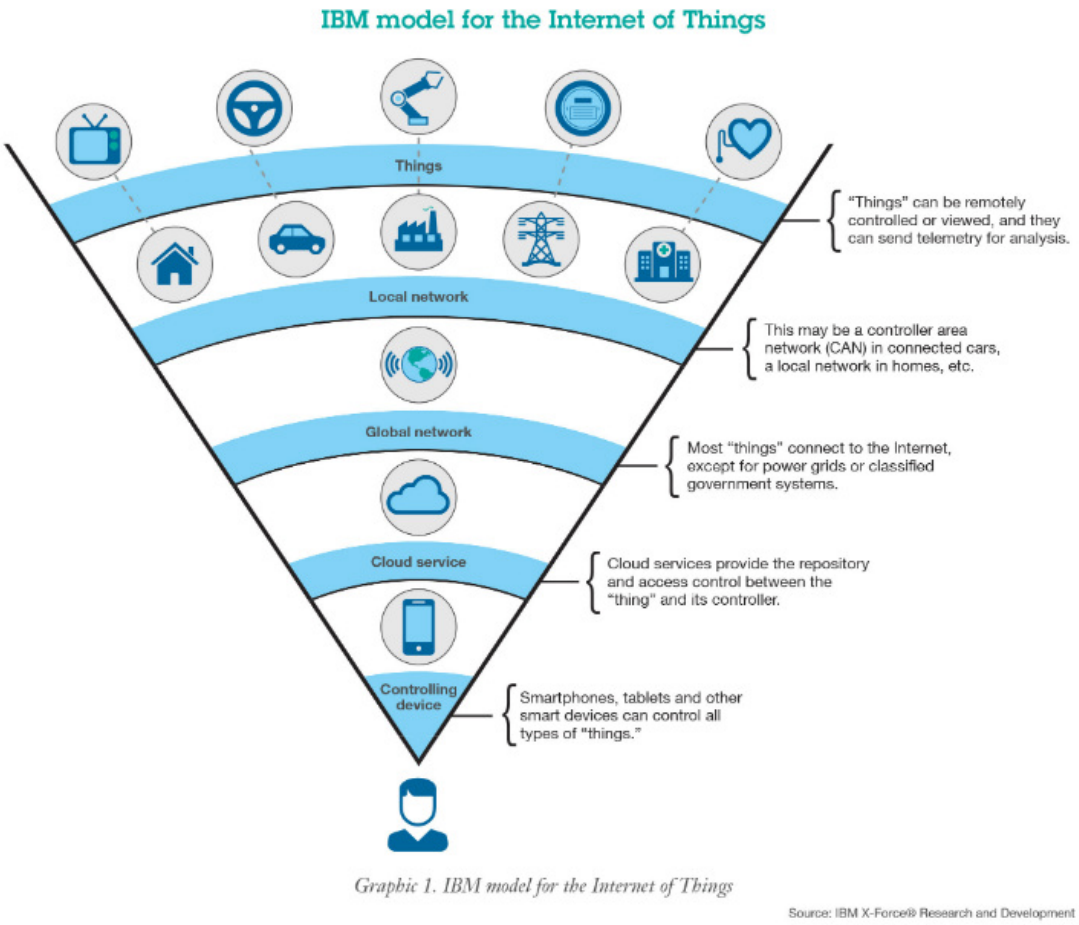


4. สิ่งประดิษฐ์ที่เกี่ยวข้องและงานวิจัยที่เกี่ยวข้อง IOT

Internet of Things (IoT) คือ "อินเตอร์เน็ตในทุกสิ่ง" หมายถึง การที่อุปกรณ์ต่าง ๆ สิ่งต่าง ๆ ได้ถูกเชื่อมโยงทุกสิ่งทุกอย่างสู่โลกอินเตอร์เน็ต ทำให้มนุษย์สามารถสั่งการควบคุมการใช้งานอุปกรณ์ต่าง ๆ ผ่านทางเครือข่ายอินเตอร์เน็ต เช่น การเปิด - ปิด อุปกรณ์เครื่องใช้ไฟฟ้า (การสั่งการเปิดไฟฟ้าภายในบ้านด้วยการเชื่อมต่ออุปกรณ์ควบคุม เช่น มือถือ ผ่านทางอินเตอร์เน็ต รถยนต์ โทรศัพท์มือถือ เครื่องมือสื่อสาร เครื่องมือทางการเกษตร อาคาร บ้านเรือน เครื่องใช้ในชีวิตประจำวันต่าง ๆ ผ่านเครือข่ายอินเตอร์เน็ต เป็นต้น

IoT มีชื่อเรียกอีกอย่างว่า M2M ย่อมาจาก Machine to Machine คือเทคโนโลยีอินเตอร์เน็ตที่เชื่อมต่ออุปกรณ์กับเครื่องมือต่าง ๆ เข้าไว้ด้วยกัน

เทคโนโลยี IoT มีความจำเป็นต้องทำงานร่วมกับอุปกรณ์ประเภท RFID และ Sensors ซึ่งเปรียบเสมือนการเติมสมองให้กับอุปกรณ์ต่าง ๆ ที่ขาดไม่คือการเชื่อมต่ออินเตอร์เน็ต เพื่อให้อุปกรณ์สามารถรับส่งข้อมูลถึงกันได้ เทคโนโลยี IoT มีประโยชน์ในหลายด้าน แต่ก็มาพร้อมกับความเสี่ยง เพราะหากระบบรักษาความปลอดภัยของอุปกรณ์ และเครือข่ายอินเตอร์เน็ตไม่ดีพอ ก็อาจทำให้มีผู้ไม่ประสงค์ดีเข้ามาขโมยข้อมูลหรือละเมิดความเป็นส่วนตัวของเราได้ ดังนั้นการพัฒนา IoT จึงจำเป็นต้องพัฒนามาตรการ และระบบรักษาความปลอดภัยไอทีควบคู่กันไปด้วย



***ภาพอธิบาย*แต่ละ NETWORK LAYERS ของ INTERNET OF THINGS โดย IBM**

ปัจจุบันมีการแบ่งกลุ่ม Internet of Things ออกตามตลาดการใช้งานเป็น 2 กลุ่มได้แก่

1.**IndustrialIoT**

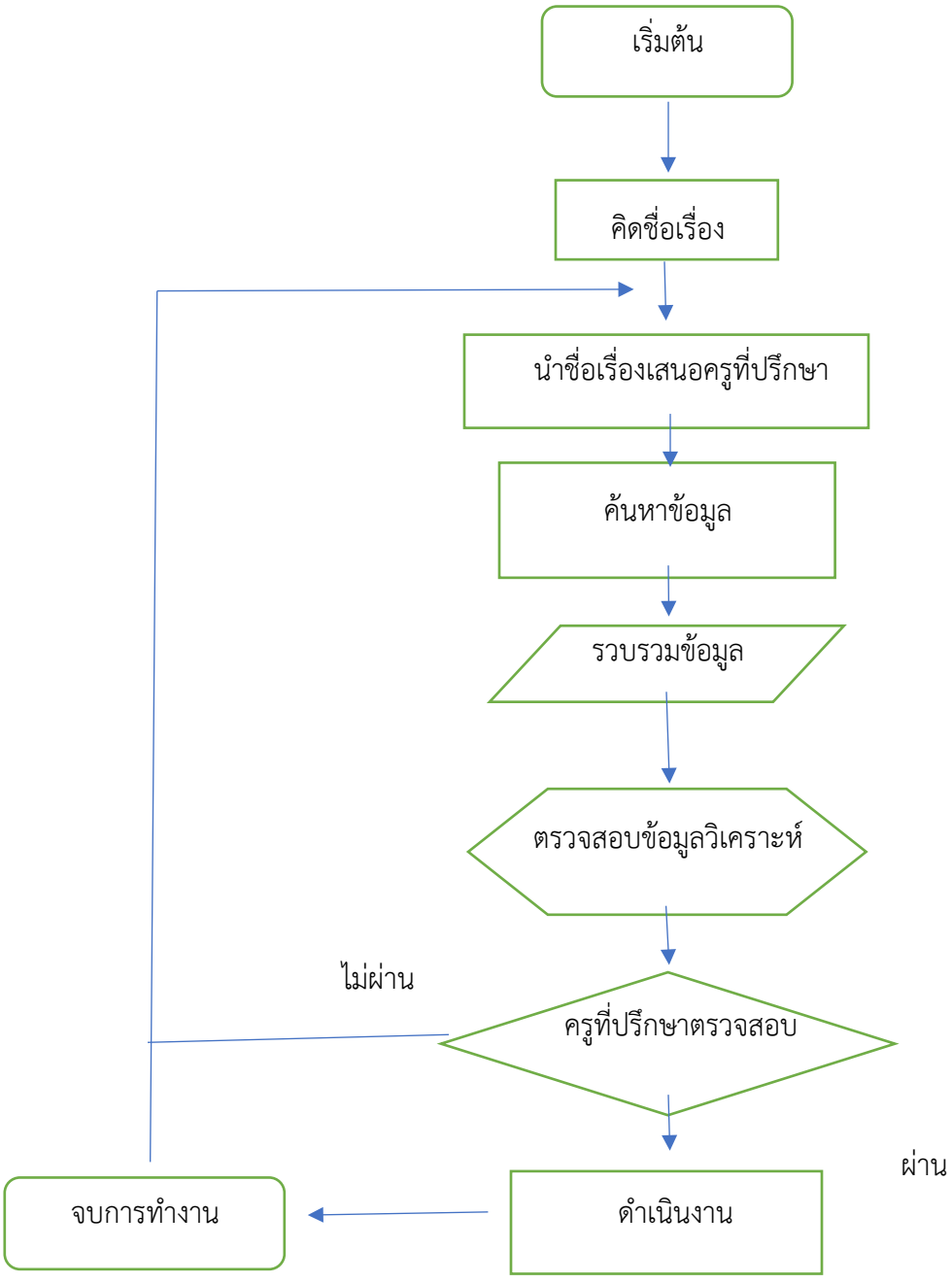
คือ แบ่งจาก local network ที่มีหลายเทคโนโลยีที่แตกต่างกันในโครงข่าย Sensor nodes โดยตัวอุปกรณ์ IoT Device ในกลุ่มนี้จะเชื่อมต่อแบบ IP network เพื่อเข้าสู่อินเตอร์เน็ต

2.**CommercialIoT**

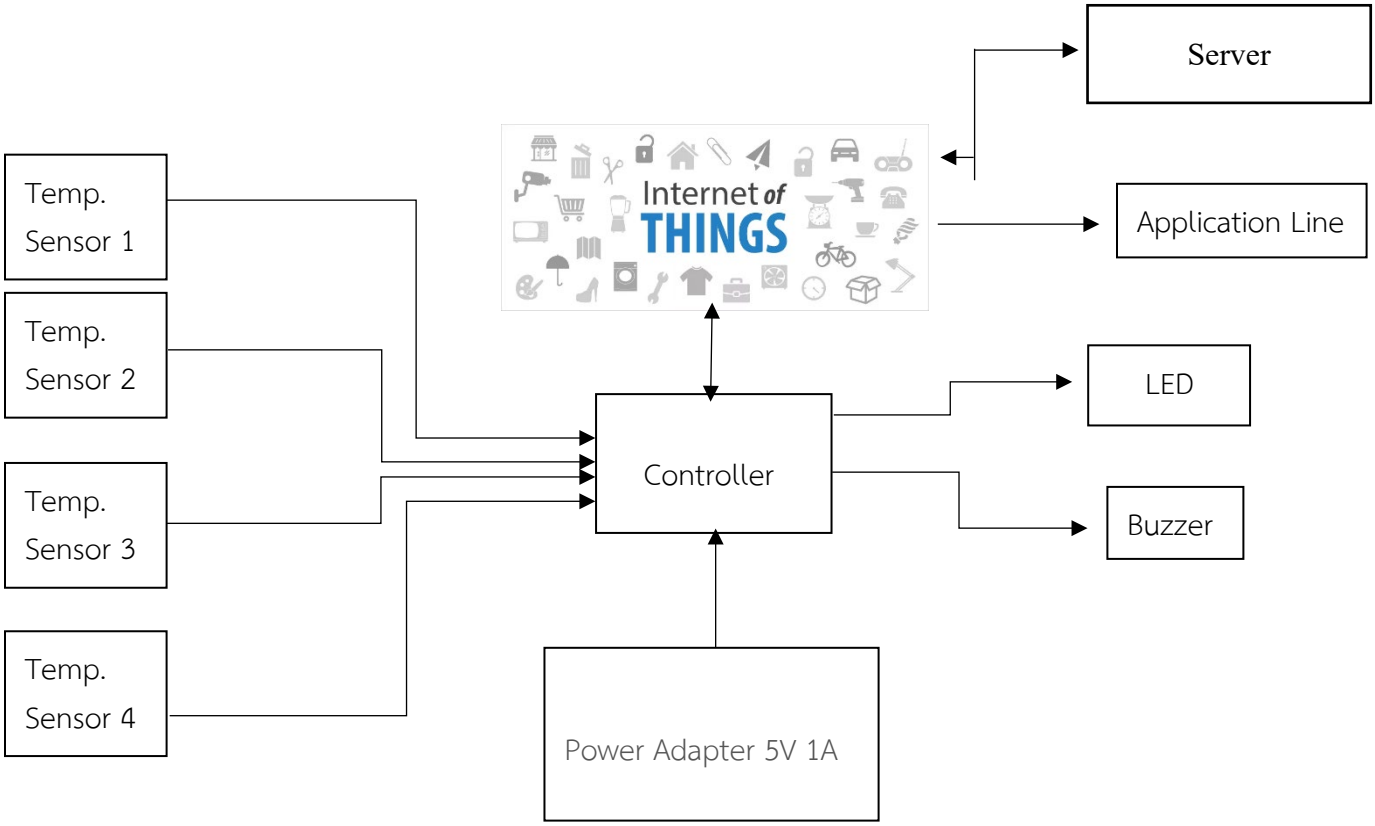
คือ แบ่งจาก local communication ที่เป็น Bluetooth หรือ Ethernet (wired or wireless) โดยตัวอุปกรณ์ IoT Device ในกลุ่มนี้จะสื่อสารภายในกลุ่ม Sensor nodes เดียวกันเท่านั้นหรือเป็นแบบ local devices เพียงอย่างเดียวอาจไม่ได้เชื่อมสู่อินเตอร์เน็ต

บทที่ 3

วิธีดำเนินการโครงการสิ่งประดิษฐ์



3.1 บล็อกไดอะแกรม อุปกรณ์ตรวจวัดอุณหภูมิ 4 จุด



บล็อกไดอะแกรม อุปกรณ์ตรวจวัดอุณหภูมิ 4 จุด

อุปกรณ์

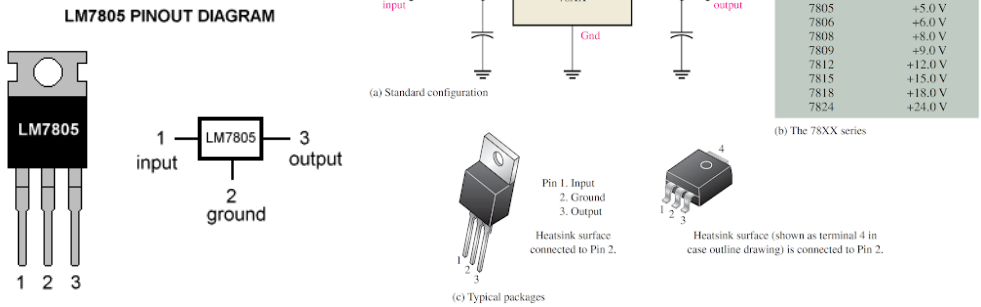
1. Power Supply 12V

คือแหล่งจ่ายพลังงานไฟฟ้าให้กับตัวอุปกรณ์หรือ Device ที่เราใช้งาน ซึ่งก็มีหลากหลายประเภท มีแบบที่เป็น linear Power Supply ก็คือพวก Transformer กับ Non-linear Power Supply หรือ Switching Power

Supply โดยเฉพาะ Switching Power Supply ที่เราจะมาแนะนำนี้ เป็นอุปกรณ์ที่จ่ายพลังงานไฟฟ้าให้กับอุปกรณ์ไฟฟ้าต่าง ๆ โดยจะทำหน้าที่แปลงแรงดันไฟฟ้ากระแสสลับ (AC) เป็นแรงดันไฟฟ้ากระแสตรง (DC) เนื่องจากอุปกรณ์ไฟฟ้าต่าง ๆ ต้องการแรงดันไฟฟ้ากระแสตรงเพื่อให้ฟังก์ชันในอุปกรณ์ทำงานได้

2. IC Voltage Regulator 7805

ไอซีที่ทำหน้าที่รักษาระดับแรงดันไฟบวกให้คงที่ ซึ่งไอซีตระกูล 78xx จะมี 3 ขา ประกอบด้วย ขาอินพุต ขาเอาต์พุต และขากราวด์



3. Controller

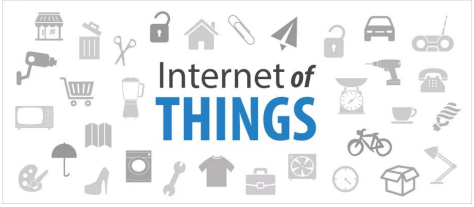
ESP8266 คือโมดูล WIFI จากจีน ที่มีความพิเศษตรงที่ตัวมันสามารถโปรแกรมลงไปได้ ทำให้สามารถนำไปใช้งานแทนไมโครคอนโทรลเลอร์ได้เลย และมีพื้นที่โปรแกรมที่มากถึง 4MB ทำให้มีพื้นที่เหลือมากในการเขียนโปรแกรมลงไปESP8266 เป็นชื่อของชิปไอซีบนบอร์ดของโมดูล ซึ่งไอซี ESP8266 ไม่มีพื้นที่โปรแกรม (flash memory) ในตัว ทำให้ต้องใช้ไอซีภายนอก (external flash memory) ในการเก็บโปรแกรม ที่ใช้การเชื่อมต่อผ่านโปรโตคอล SPI ซึ่งสาเหตุนี้เองทำให้โมดูล ESP8266 มีพื้นที่โปรแกรมมากกว่าไอซีไมโครคอนโทรลเลอร์เบอร์อื่น ๆ ESP8266 ทำงานที่แรงดันไฟฟ้า 3.3V - 3.6V การนำไปใช้งานร่วมกับเซ็นเซอร์อื่น ๆ ที่ใช้แรงดัน 5V ต้องใช้

วงจรแบ่งแรงดันมาช่วย เพื่อไม่ให้โมดูลพังเสียหาย กระแสที่โมดูลใช้งานสูงสุดคือ 200mA ความถี่คริสตอล 40MHz

ขาของโมดูล ESP8266 แบ่งได้ดังนี้

- VCC เป็นขาสำหรับจ่ายไปเข้าเพื่อให้โมดูลทำงานได้ ซึ่งแรงดันที่ใช้งานได้คือ 3.3 - 3.6V
- GND
- Reset และ CH_PD (หรือ EN) เป็นขาที่ต้องต่อเข้าไฟ + เพื่อให้โมดูลสามารถทำงานได้ ทั้ง 2 ขานี้สามารถนำมาใช้รีเซ็ตโมดูลได้เหมือนกัน แตกต่างตรงที่ขา Reset สามารถลอยไว้ได้ แต่ขา CH_PD (หรือ EN) จำเป็นต้องต่อเข้าไป + เท่านั้น เมื่อขานี้ไม่ต่อเข้าไฟ + โมดูลจะไม่ทำงานทันที
- GPIO เป็นขาดิจิตอลอินพุต / เอาต์พุต ทำงานที่แรงดัน 3.3V
- GPIO15 เป็นขาที่ต้องต่อลง GND เท่านั้น เพื่อให้โมดูลทำงานได้
- GPIO0 เป็นขาทำหรับการเลือกโหมดทำงาน หากนำขานี้ลง GND จะเข้าโหมดโปรแกรม หากลอยไว้ หรือนำเข้าไฟ + จะเข้าโหมดการทำงานปกติ
- ADC เป็นขาอนาล็อกอินพุต รับแรงดันได้สูงสุดที่ 1V ขนาด 10 บิต การนำไปใช้งานกับแรงดันที่สูงกวต้องใช้วงจรแบ่งแรงดันเข้าช่วย

4. **Internet of Things (IoT) คือ** "อินเตอร์เน็ตในทุกสิ่ง" หมายถึง การที่อุปกรณ์ต่าง ๆ สิ่งต่าง ๆ ได้ถูกเชื่อมโยงทุกสิ่งทุกอย่างสู่โลกอินเตอร์เน็ต ทำให้มนุษย์สามารถสั่งการควบคุมการใช้งานอุปกรณ์ต่าง ๆ ผ่านทางเครือข่ายอินเตอร์เน็ต เช่น การเปิด - ปิด อุปกรณ์เครื่องใช้ไฟฟ้า (การสั่งการเปิดไฟฟ้าภายในบ้านด้วยการเชื่อมต่ออุปกรณ์ควบคุม เช่น มือถือ ผ่านทางอินเตอร์เน็ต รถยนต์ โทรศัพท์มือถือ เครื่องมือสื่อสาร เครื่องมือทางการเกษตร อาคาร บ้านเรือน เครื่องใช้ในชีวิตประจำวันต่าง ๆ ผ่านเครือข่ายอินเตอร์เน็ต เป็นต้น



5.Application Line

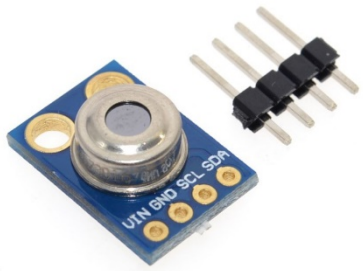
ไลน์ (Line) คือ Application สำหรับการสื่อสารยอดนิยม เนื่องจากมีความสามารถหลากหลาย และทำงานได้บนหลากหลายอุปกรณ์ ไม่ว่าจะเป็นสมาร์ทโฟน, แท็บเล็ต หรือแม้กระทั่งบนเครื่องคอมพิวเตอร์ สำหรับความสามารถเด่น ๆ ที่ทำให้ Line มีความแตกต่างจาก Application สื่อสารอื่น ๆ ก็คือรูปภาพตัวการ์ตูนสื่ออารมณ์ที่เรียกว่าสติ๊กเกอร์ ประโยชน์ของมันคือช่วยให้ลดปริมาณการพิมพ์ข้อความ และช่วยสร้างความแปลกใหม่ในการสนทนาได้เป็นอย่างดี ทำให้ในปัจจุบันนี้ Line ได้รับความนิยมเป็นอย่างมาก

6. Infrared Sensor

เป็นเซนเซอร์วัดอุณหภูมิแบบไร้การสัมผัส โดยใช้หลักการแปลงแสงอินฟาเรดที่ส่งออกจากตัววัตถุซึ่งจะมีสีแตกต่างกัน ให้เป็นค่าอุณหภูมิโมดูลวัดอุณหภูมิแบบอินฟาเรดไร้การสัมผัส ใช้ชิฟ MLX90614ESF สำหรับ Arduino ไฟเลี้ยง 3V-5V เชื่อมต่อแบบ I2C ใช้สายเพียง 2 เส้นในการควบคุม สามารถวัดอุณหภูมิที่เป้าหมายแบบไร้การสัมผัสที่ -70 ถึง 380 องศาเซลเซียส และยังสามารถวัดอุณหภูมิของสิ่งแวดล้อมได้ที่ -40 ถึง 125 องศาเซลเซียส ความละเอียดของอุณหภูมิที่วัดได้ 0.02 องศาเซลเซียส

1. ต่อวงจร Arduino GY-906 MLX90614 กับบอร์ด Arduino Uno ดังนี้

ขา VCC - 3.3-5V
ขา GND – GND
ขา SDA – D1
ขา SCL – D2



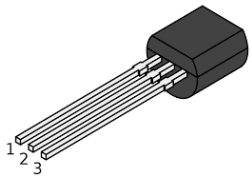
7. LED

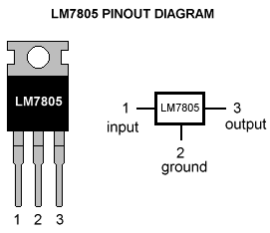
LED คือไดโอดเปล่งแสง ย่อมาจากคำว่า(Light-Emitting Diode) ซึ่งสามารถเปล่งแสงออกมาได้แสงที่เปล่งออกมาประกอบด้วยคลื่นความถี่เดียวและ เฟสต่อเนื่องกัน ซึ่งต่างกับแสงธรรมดาที่ตาคนมองเห็น โดย *หลอด LED* สามารถเปล่งแสงได้เมื่อจ่ายกระแสไฟฟ้าเข้าเพียงเล็กน้อยเท่านั้น และประสิทธิภาพในการให้แสงสว่างก็ยังดีกว่าหลอดไฟขนาดเล็กทั่ว ๆ ไป **LED** โดยทั่วไปมี 2 ชนิดใหญ่ ๆ คือ*LED* ชนิดที่ตาคนเห็นได้ กับชนิดที่ตาคนมองไม่เห็นต้องใช้ทรานซิสเตอร์มาเป็นตัวรับแสงแทนตาคน

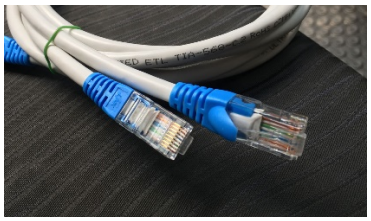
8. Buzzer

Buzzer บลัซเซอร์ คือ ลำโพงแบบแม่เหล็กหรือ แบบเปียโซที่มีวงจรกำเนิดความถี่ (oscillator) อยู่ภายในตัว ใช้ไฟเลี้ยง 3.3 - 5V สามารถสร้างเสียงเตือนหรือส่งสัญญาณที่เป็นรูปแบบต่าง ๆ เราอาจจะเคยได้ยินเสียงบลัซเซอร์อยู่บ่อย ๆ เช่น เสียง ปี๊บที่อยู่ในคอมพิวเตอร์ก็ใช้บลัซเซอร์ในการส่งสัญญาณให้ทราบสถานะของคอมพิวเตอร์ให้ทราบว่ามีปัญหาอะไร

รายการอุปกรณ์

ลำดับ	รายการ	หน่วย	จำนวน	ระยะเวลาการใช้	รูปภาพ
1	Buzzer H10 (REST)	อัน	1	เบิดขาด	
2	ESP 8266	อัน	3	เบิกขาด	
3	LED 5 mm สีแดง	ตัว	4	เบิกขาด	
4	ทรานซิสเตอร์ TO-92-3	ตัว	1	เบิกขาด	

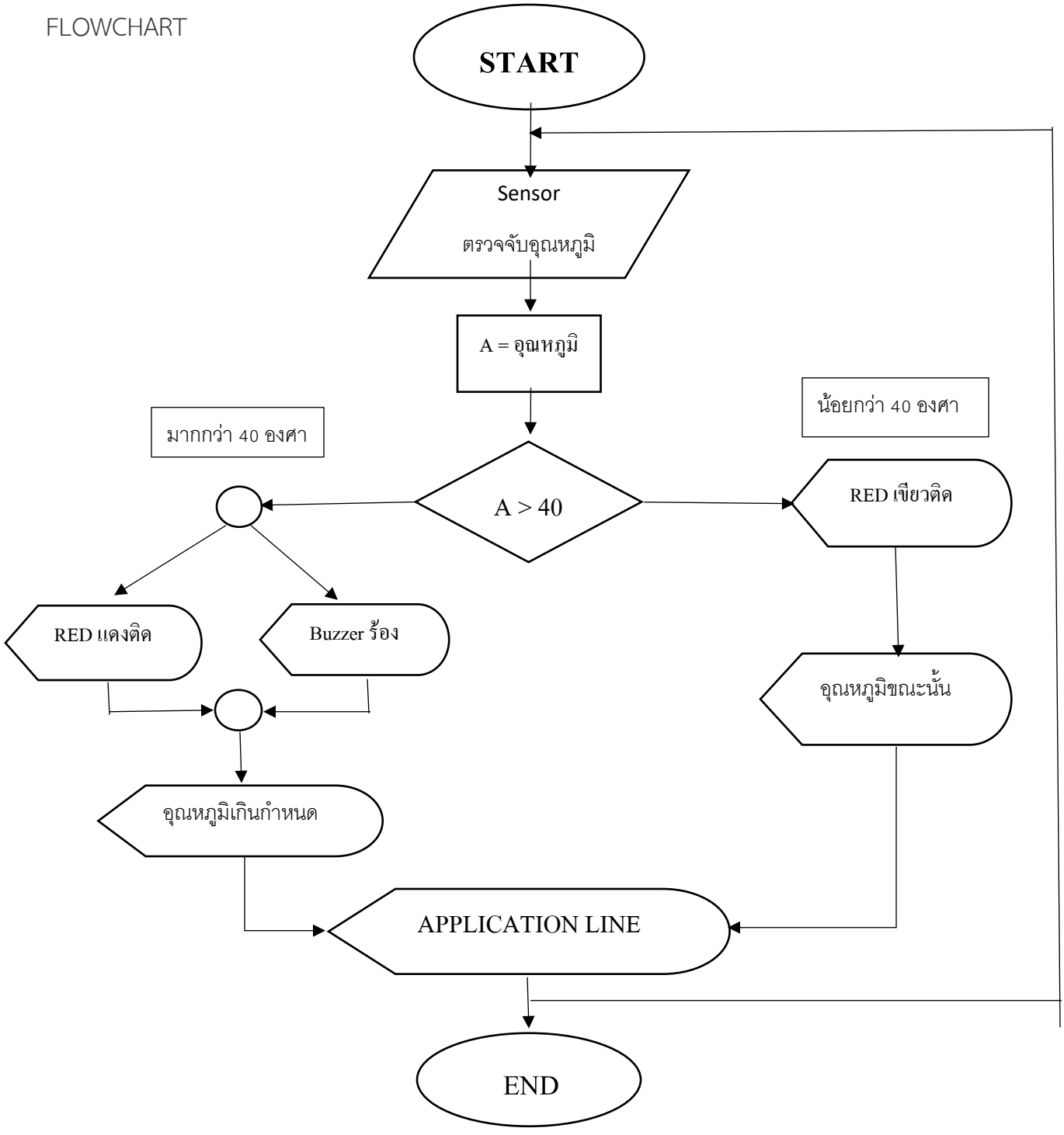
ลำดับ	รายการ	หน่วย	จำนวน	ระยะเวลาการใช้	รูปภาพ
5	GY-906 เซ็นเซอร์ IR Temp.A1,A153 (REST)	อัน	4	เบิดขาด	
6	ไขควงวัดไฟ (WA)	อัน	1	เบิกยืม	
7	เทปพันสาย (REST)	ม้วน	1	เบิกยืม	
8	Regulate 5v	ตัว	1	เบิกขาด	

ลำดับ	รายการ	หน่วย	จำนวน	ระยะเวลาการใช้	รูปภาพ
9	สายแลน	เมตร	35	เบิดขาด	
10	Power Supply 12V	อัน	1	เบิกยืม	
11	บัดกรี	อัน	1	เบิกยืม	
12	ไขควงปากแบน	อัน	1	เบิกยืม	

ลำดับ	รายการ	หน่วย	จำนวน	ระยะเวลาการใช้	รูปภาพ

13	ไขควงปากแฉก	อัน	1	เบิกยืม	
14	คีมจับปลายแหลม	อัน	1	เบิกยืม	
15	คีมตัด	อัน	1	เบิกยืม	
16	ที่ดูดตะกั่วแบบปั๊ม	อัน	1	เบิกยืม	
17	เวอร์เนียคาลิปเปอร์	อัน	1	เบิกยืม	

FLOWCHART


START
Sensor
ตรวจจับอุณหภูมิ
A = อุณหภูมิ
มากกว่า 40 องศา
น้อยกว่า 40 องศา
A > 40
RED เขียวติด
RED แดงติด
Buzzer ร้อง
อุณหภูมิขณะนั้น
อุณหภูมิเกินกำหนด
APPLICATION LINE
END

แผนการดำเนินงาน

รายการปฏิบัติงาน	ต.ค. 62				พ.ย. 62				ธ.ค. 62				ม.ค. 63				ก.พ. 63				มี.ค. 63			
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
ขอครูเป็นที่ปรึกษา																								
ค้นคว้าหาข้อมูล																								
เสนอชื่อโครงงานกับครูที่ปรึกษา																								
กลั่นกรองโครงงาน																								
- ศึกษาข้อมูลอุปกรณ์ - ตรวจสอบราคา - เสนอรายการอุปกรณ์																								
ขออนุมัติจัดทำโครงงาน																								
ออกแบบวงจร																								
จัดทำโครงงาน																								
เขียนโปรแกรม																								

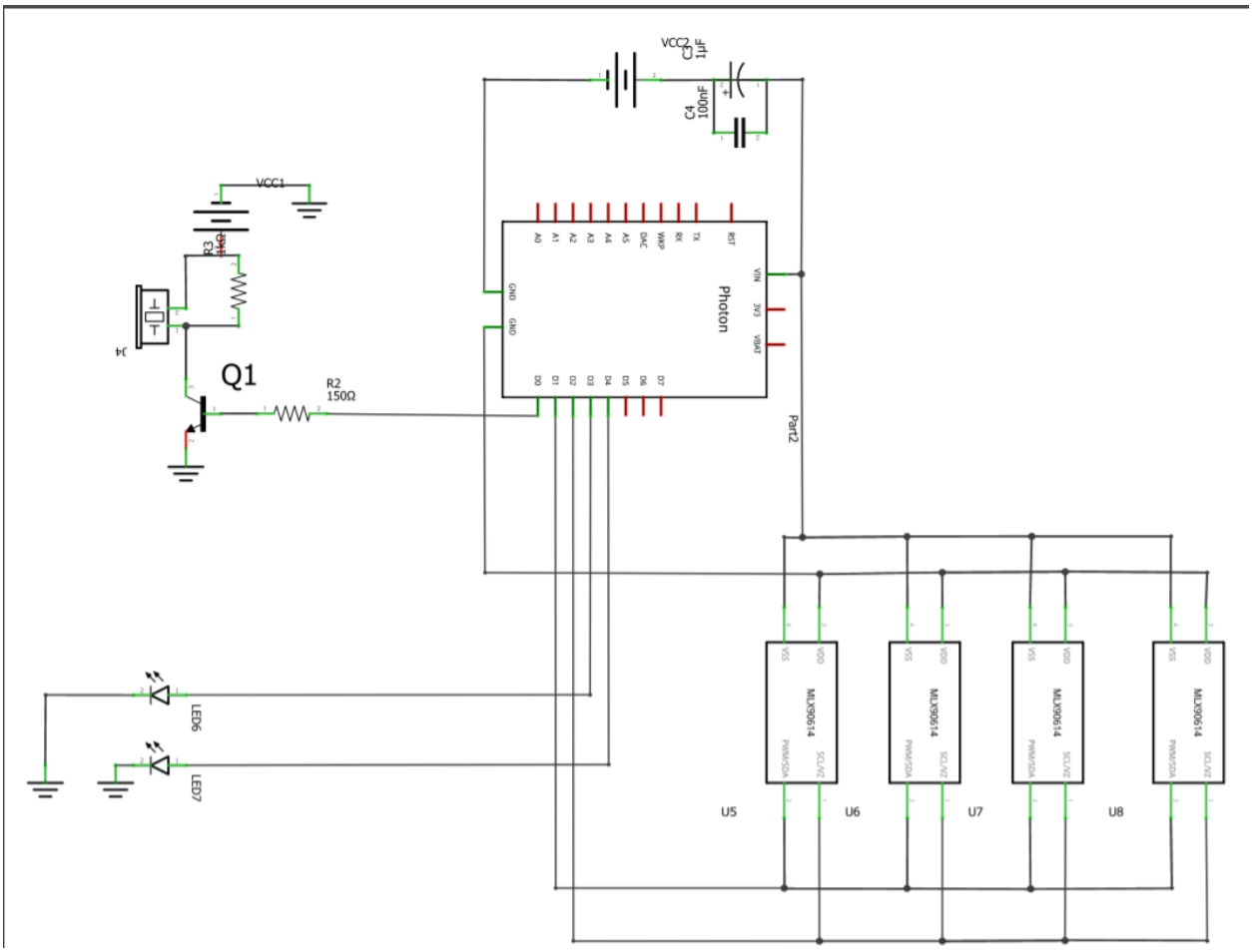
รายการปฏิบัติติงาน	ต.ค. 62	พ.ย. 62	ธ.ค. 62	ม.ค. 63	ก.พ. 63	มี.ค. 63

	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
ทดสอบการทำงาน																								
ตรวจสอบการทำงาน																								
ทำเคสใส่อุปกรณ์																								
ปรับปรุงแก้ไข																								
จัดทำเอกสารโครงงาน																								
บทที่ 1																								
บทที่ 2																								
บทที่ 3																								
บทที่ 4																								
บทที่ 5																								
PowerPoint																								
แผ่นพับ																								
นำเสนอโครงงาน																								
ส่งโครงงานและเอกสาร																								
จัดทำนิเทศการโครงงาน																								

การทำงานของอุปกรณ์ตรวจวัดอุณหภูมิ 4 จุด

เมื่อ Infrared Sensor จับอุณหภูมิที่สูงขึ้นเกินกว่าที่ตั้งโปรแกรมไว้ ESP8266 จะประมวณผลคำสั่งส่งให้ CLOUD เพื่อให้ LINE แจ้งเตือน “ ว่าอุณหภูมิเกินกำหนด ” พร้อมกับ LED เปล่งแสงส่วน BUZZER จะร้องแจ้งเตือน แต่ถ้าหากอุณหภูมิไม่สูงเกินกว่าที่ตั้งไว้เซ็นจะตรวจจับอุณหภูมิห้องในขณะนั้นและส่งการแจ้งเตือนเข้า LINE มาเป็นอุณหภูมิห้องในขณะนั้นทุก ๆ 1 นาที

วงจร Wiring Diagram



บทที่ 4

ผลการทดลอง

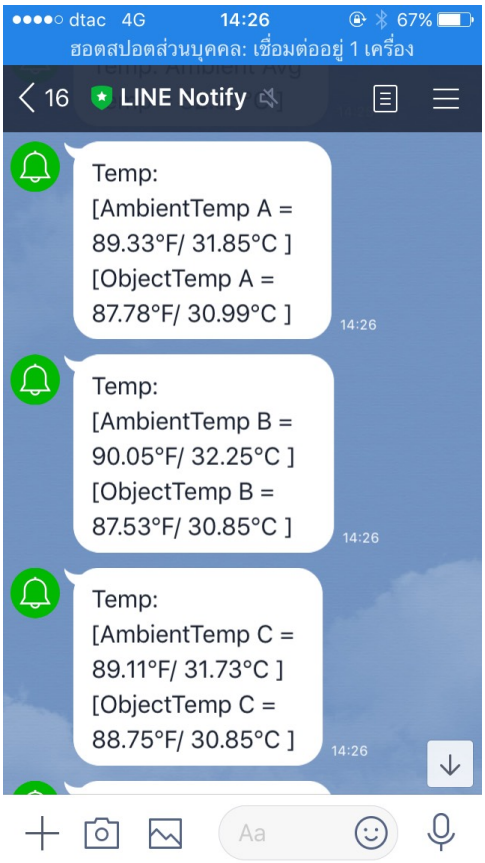
จากการศึกษาข้อมูลและสร้างโครงงานสิ่งประดิษฐ์ในครั้งนี้

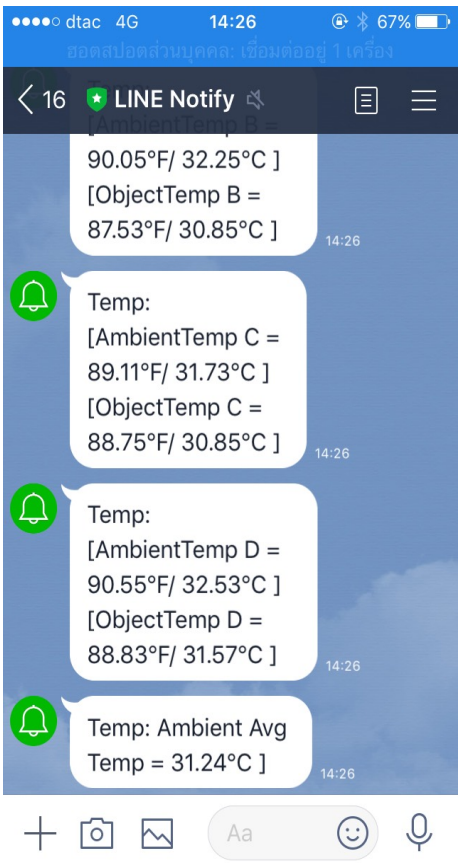
ผู้จัดทำได้ตรวจจับอุณหภูมิของห้อง จ.

1	ความสูงของห้องจากพื้นห้อง (เมตร)	ครั้ง/ชั่วโมง									
		เซ็นเซอร์ตัวที่ 1		เซ็นเซอร์ตัวที่ 2		เซ็นเซอร์ตัวที่ 3		เซ็นเซอร์ตัวที่ 4		ค่าเฉลี่ยทั้งหมด	
		องศาเซลเซียส/นาที		องศาเซลเซียส/นาที		องศาเซลเซียส/นาที		องศาเซลเซียส/นาที		องศาเซลเซียส/นาที	
28 เมตร	0.5 เมตร	31.85°C	1	32.25°C	1	31.73°C	1	32.53°C	1	32.09°C	1
	1 เมตร	31.83°C	1	32.25°C	1	31.69°C	1	32.01°C	1	31.94°C	1
	1.5 เมตร	32.09°C	1	32.45°C	1	31.91°C	1	32.03°C	1	32.12°C	1
	2 เมตร	32.09°C	1	32.43°C	1	31.89°C	1	32.15°C	1	32.14°C	1

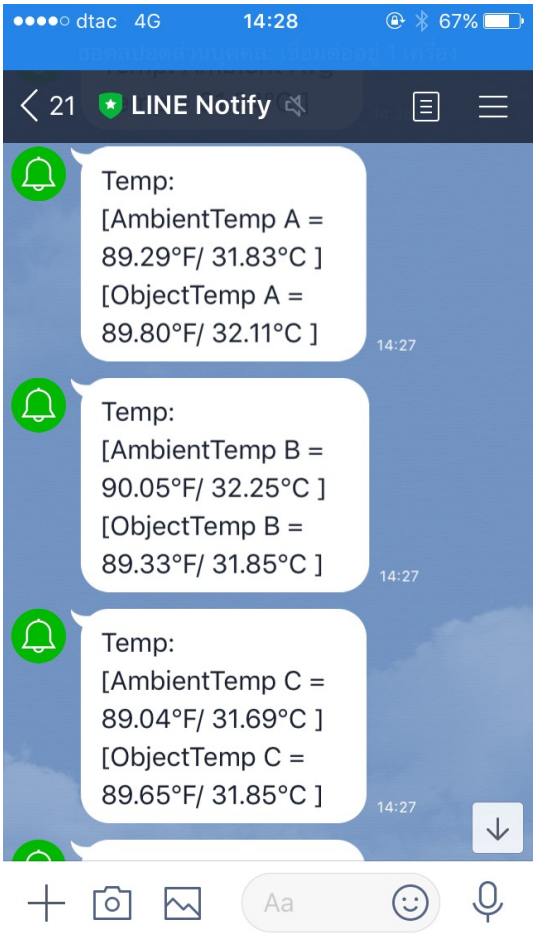
ผลที่ได้ การทดลองรอบห้อง 28 เมตร สูง 0.5 เมตรถึง 2.5 เมตร เซ็นเซอร์จำนวน 4 ตัว สามารถส่งข้อมูลเฉลี่ยได้ 32.07 องศาเซลเซียส/นาที

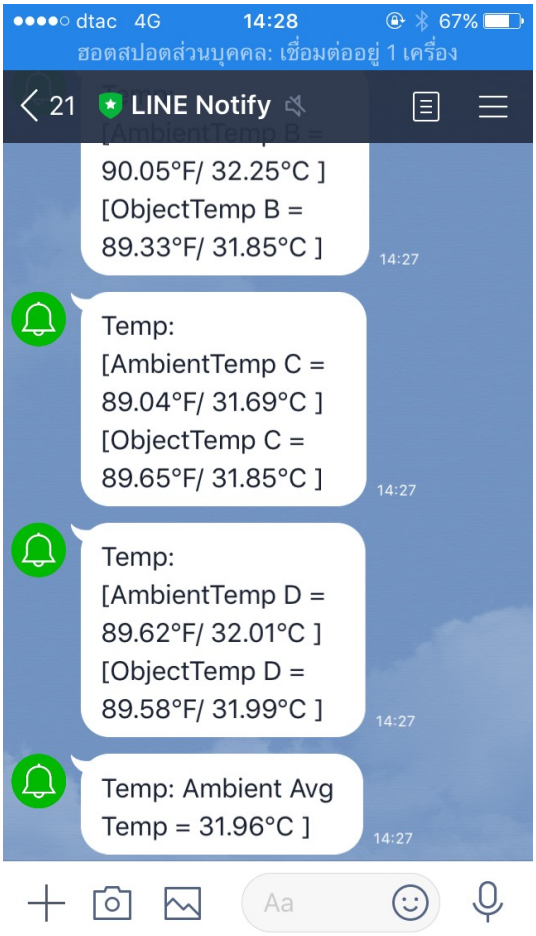
การทดลองที่ 0.5 เมตร


dtac 4G 14:26 67%
ฮอตสปอตส่วนบุคคล: เชื่อมต่ออยู่ 1 เครื่อง
16 LINE Notify
Temp:
[AmbientTemp A =
89.33°F/ 31.85°C]
[ObjectTemp A =
87.78°F/ 30.99°C]
14:26
Temp:
[AmbientTemp B =
90.05°F/ 32.25°C]
[ObjectTemp B =
87.53°F/ 30.85°C]
14:26
Temp:
[AmbientTemp C =
89.11°F/ 31.73°C]
[ObjectTemp C =
88.75°F/ 30.85°C]
14:26
Aa

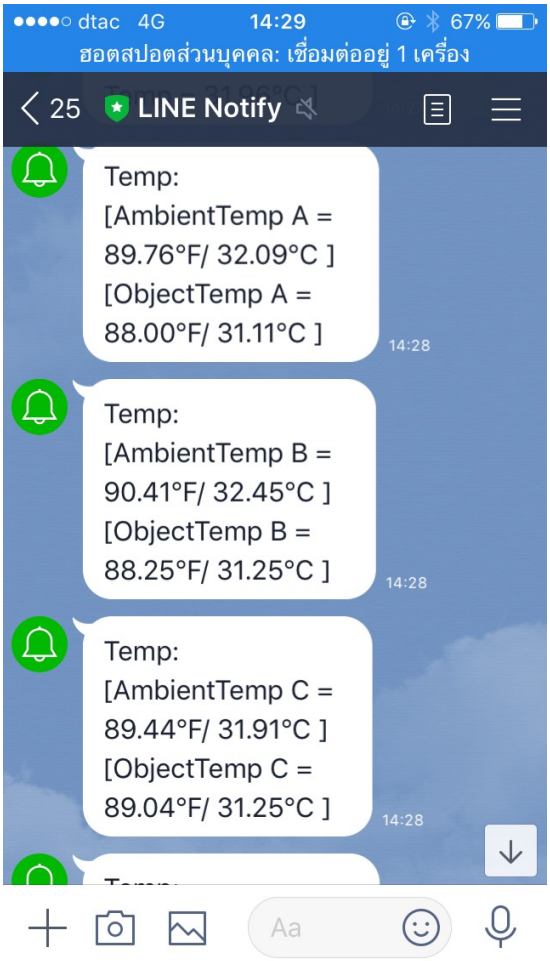

dtac 4G 14:26 67%
16 LINE Notify
90.05°F/ 32.25°C]
[ObjectTemp B =
87.53°F/ 30.85°C]
14:26
Temp:
[AmbientTemp C =
89.11°F/ 31.73°C]
[ObjectTemp C =
88.75°F/ 30.85°C]
14:26
Temp:
[AmbientTemp D =
90.55°F/ 32.53°C]
[ObjectTemp D =
88.83°F/ 31.57°C]
14:26
Temp: Ambient Avg
Temp = 31.24°C]
14:26
Aa

การทดลองที่ 1 เมตร

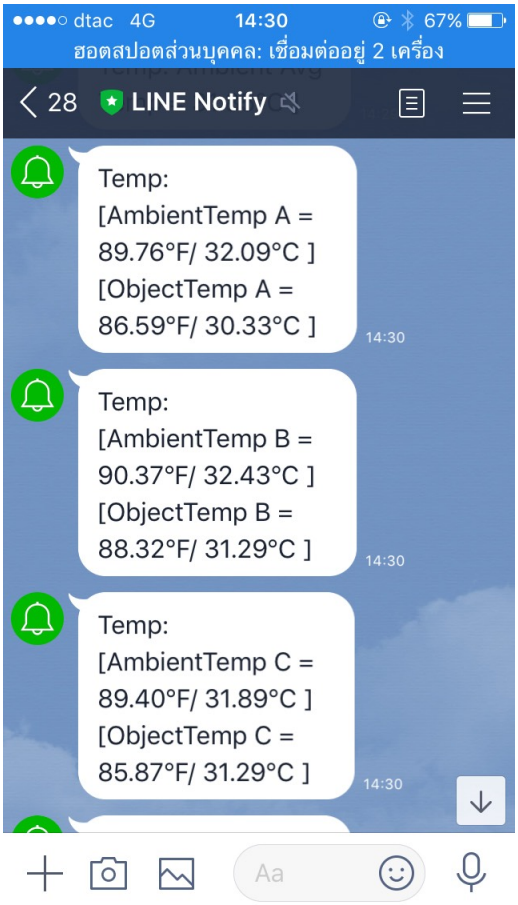




การทดลองที่ 1.5 เมตร



การทดลองที่ 2 เมตร


dtac 4G 14:30 67%
ฮอตสปอตส่วนบุคคล: เชื่อมต่ออยู่ 2 เครื่อง
28 LINE Notify
Temp:
[AmbientTemp A =
89.76°F/ 32.09°C]
[ObjectTemp A =
86.59°F/ 30.33°C]
14:30
Temp:
[AmbientTemp B =
90.37°F/ 32.43°C]
[ObjectTemp B =
88.32°F/ 31.29°C]
14:30
Temp:
[AmbientTemp C =
89.40°F/ 31.89°C]
[ObjectTemp C =
85.87°F/ 31.29°C]
14:30
Aa

dtac 4G 14:30 67%
ฮอตสปอตส่วนบุคคล: เชื่อมต่ออยู่ 2 เครื่อง
28 LINE Notify
90.37°F/ 32.43°C]
[ObjectTemp B =
88.32°F/ 31.29°C]
14:30
Temp:
[AmbientTemp C =
89.40°F/ 31.89°C]
[ObjectTemp C =
85.87°F/ 31.29°C]
14:30
Temp:
[AmbientTemp D =
89.87°F/ 32.15°C]
[ObjectTemp D =
86.56°F/ 30.31°C]
14:30
Temp: Ambient Avg
Temp = 30.54°C]
14:30
Aa

บทที่ 5

สรุปผล อภิปรายผลและข้อเสนอแนะ

ในการจัดทำโครงงาน อุปกรณ์ตรวจจับอุณหภูมิ 4 จุด นี้สามารถสรุปผลการดำเนินงานและข้อเสนอแนะได้ดังนี้

5.1 สรุปผลและอภิปรายผล

5.1.1. วัตถุประสงค์ของโครงงาน

5.1.1.1. สร้างระบบแจ้งเตือน ไฟไหม้ ผ่านระบบสื่อสาร IOT ได้

5.1.1.2 สามารถแจ้งเตือนผ่าน Application Line โดยอาศัยระบบ Internet

5.1.1.3 สามารถระงับเหตุไฟไหม้ได้ในทันทีและรวดเร็ว

5.1.1.4 สามารถนำผลงานชิ้นนี้ไปใช้ในการป้องกันไฟไหม้ในห้องหรือภายในตัวอาคาร

5.1.2. วัสดุ อุปกรณ์ เครื่องมือหรือโปรแกรมที่ใช้ในการพัฒนา

5.1.2.1. บอร์ด Node MCU ESP8266

5.1.2.2. Buzzer H10

5.1.2.3. GY-906 เซ็นเซอร์ IR Temp.A1, A153

5.1.2.4. สายแลน

5.1.2.5. Power Supply 5V 1 A

5.1.2.6. คาปาซิเตอร์

5.2. สรุปผลการดำเนินงาน

ในการจัดทำโครงงานเพื่อนำไปปฏิบัติงานจริงในเรื่อง ระบบแจ้งเตือนนิรภัยด้วยระบบสื่อสาร IOT (IOT Warning And Communication Rescue) ผู้จัดทำโครงงานมีวัตถุประสงค์เพื่อการติดตามและประเมินผลการทำงาน สามารถนำไปพัฒนาต่อไปในการนำไปใช้ในงานการแจ้งเตือนเมื่อเกิดเหตุอัคคีภัย ดังนั้น ผลการดำเนินงานหรือข้อมูลทั้งหมดมี ผลการดำเนินงานดังนี้

5.2.1. เมื่อ Infrared Sensor จับอุณหภูมิที่เกินกว่าที่ตั้งโปรแกรมไว้ Node MCU ESP8266 จะประมวณผลคำสั่งส่งไปให้ IoT เพื่อให้ LINE แจ้งเตือนว่า “ อุณหภูมิเกินกำหนด ” พร้อมกับ LED เปล่งแสงสีแดงส่วน BUZZER จะร้องแจ้งเตือน

5.3. ปัญหาและอุปสรรคในการจัดทำโครงการ

5.3.1. เพื่อนไม่ให้ความร่วมมือ

5.3.2. เรื่องการเขียนโปรแกรม

5.3.3. เรื่องอุปกรณ์เซนเซอร์ วัดระยะได้น้อย

5.3.4. มีการเปลี่ยนแปลงแผนโครงงานตามความเหมาะสมของอุปกรณ์ตลอดเวลา

5.3.5. ขาดความรู้และความเข้าใจเรื่องอุปกรณ์ อิเล็กทรอนิกส์

5.3.6. เรื่องการออกแบบโครงสร้างรูปลักษณ์

5.4. ข้อเสนอแนะ

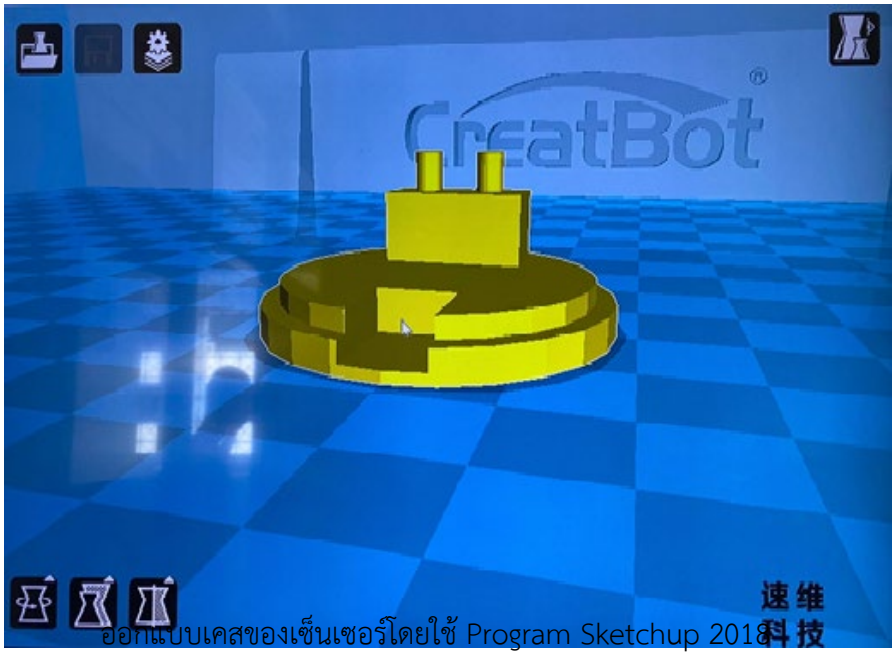
5.4.1. มีการใช้ Capacitor เพื่อลดทอนสัญญาณรบกวนที่มาจากแหล่งจ่าย

5.4.2. ในการวัดแหล่งจ่ายทุกครั้งให้ใช้สายจั๊มต่อสายออกมาจากวงจรเพื่อกันการช๊อต

5.4.3. ใช้โปรแกรม Sketch up ในการออกแบบเคสเพื่อลดต้นทุนในการทำงาน

5.4.4. ข้อจำกัดในการใช้สายไฟหรือสายแลนด์

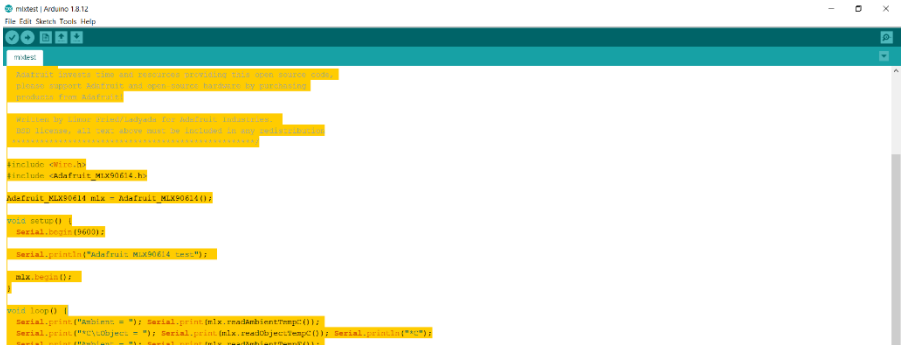
ภาคผนวก



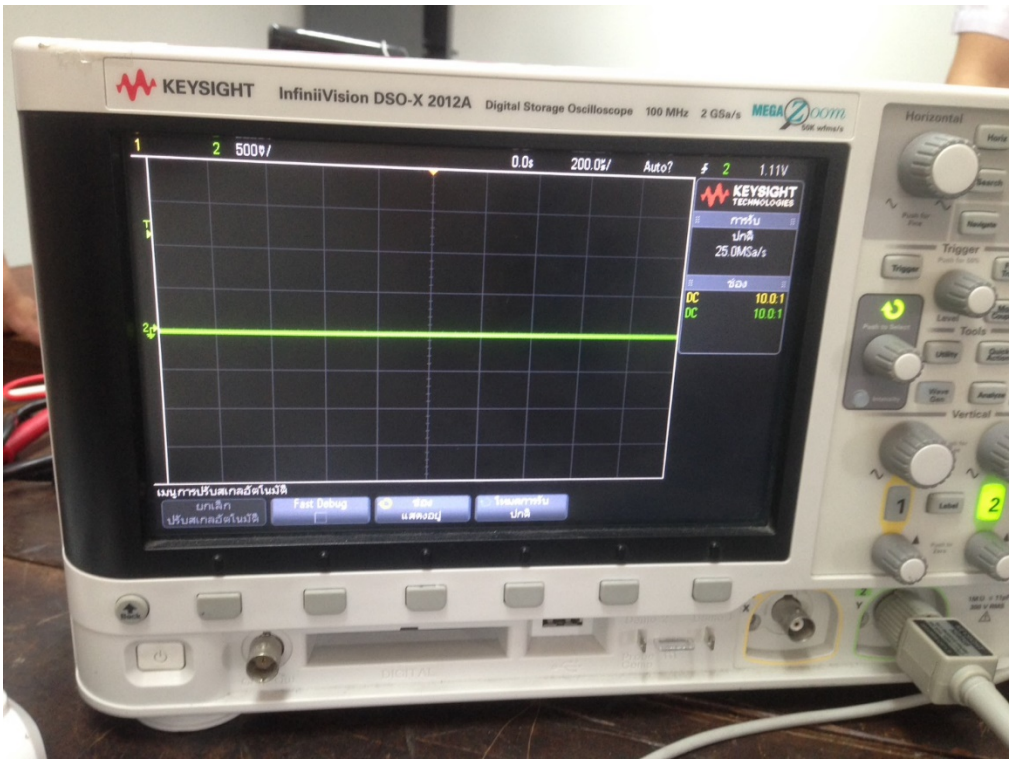
ออกแบบเคสของเซ็นเซอร์โดยใช้ Program Sketchup 2018

ชิ้นงานที่สร้างจาก เครื่องพิมพ์ 3D Printer

เราเขียน Code Client โดยมีครูที่ปรึกษาคอยแนะนำ



เราใช้ Oscilloscope ในการวัดสัญญาณรบกวน



แหล่งจ่ายที่ผ่านการกรองสัญญาณแล้วโดยใช้ Capacitor

เอกสารอ้างอิง

การทำรายงาน 5 บท

https://www.slideshare.net/AekapojPoosathan/5-31018403

ค้นหาซื้อขายอุปกรณ์

https://www.arduinoall.com/?gclid=Cj0KCQjw6sHzBRCbARIsAF8FMpV_kdN8Tj4l4IXohmFdnPjTZTDNQYIzsa2K4oGgKBItQTs__5hNxYwaAvH-EALw_wcB

ติดตั้งโปรแกรม Sketchup Pro 2018

https://sketchup.th.downloadastro.com/

ขั้นตอนการติดตั้ง Arduino IDE กับ ESP8266

https://playground.cmmakerclub.com/2015/06/esp8266/%E0%B9%80%E0%B8%A3%E0%B8%B4%E0%B9%88%E0%B8%A1%E0%B8%95%E0%B9%89%E0%B8%99%E0%B9%83%E0%B8%8A%E0%B9%89%E0%B8%87%E0%B8%B2%E0%B8%99-esp8266-%E0%B8%9C%E0%B9%88%E0%B8%B2%E0%B8%99-arduino-ide/