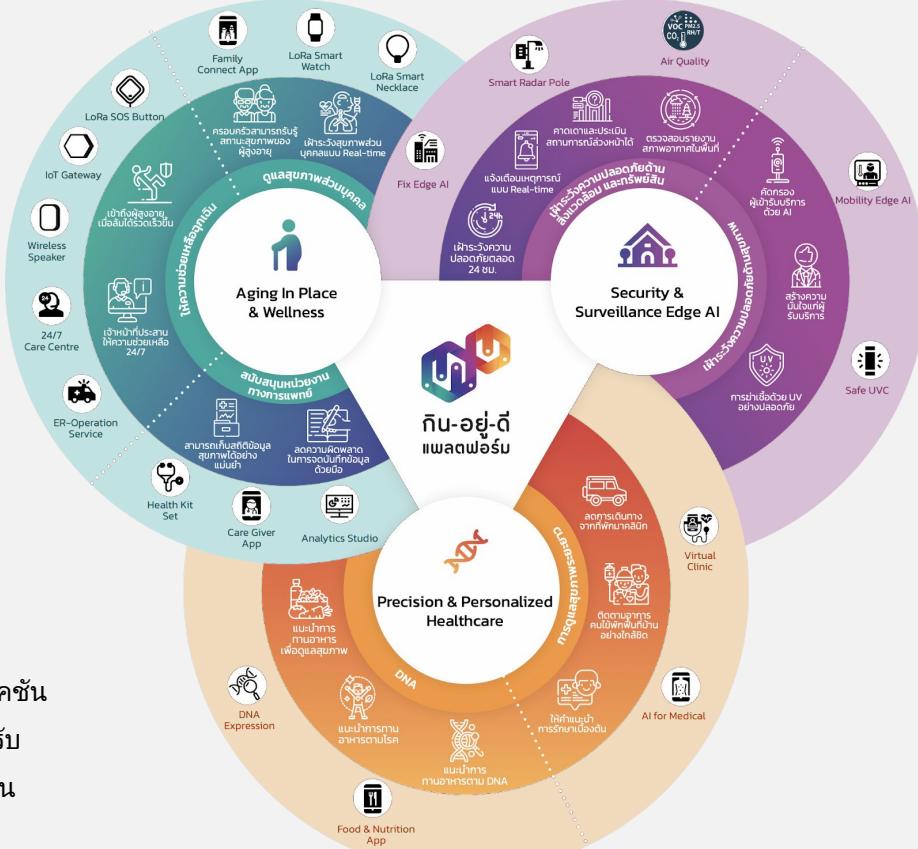
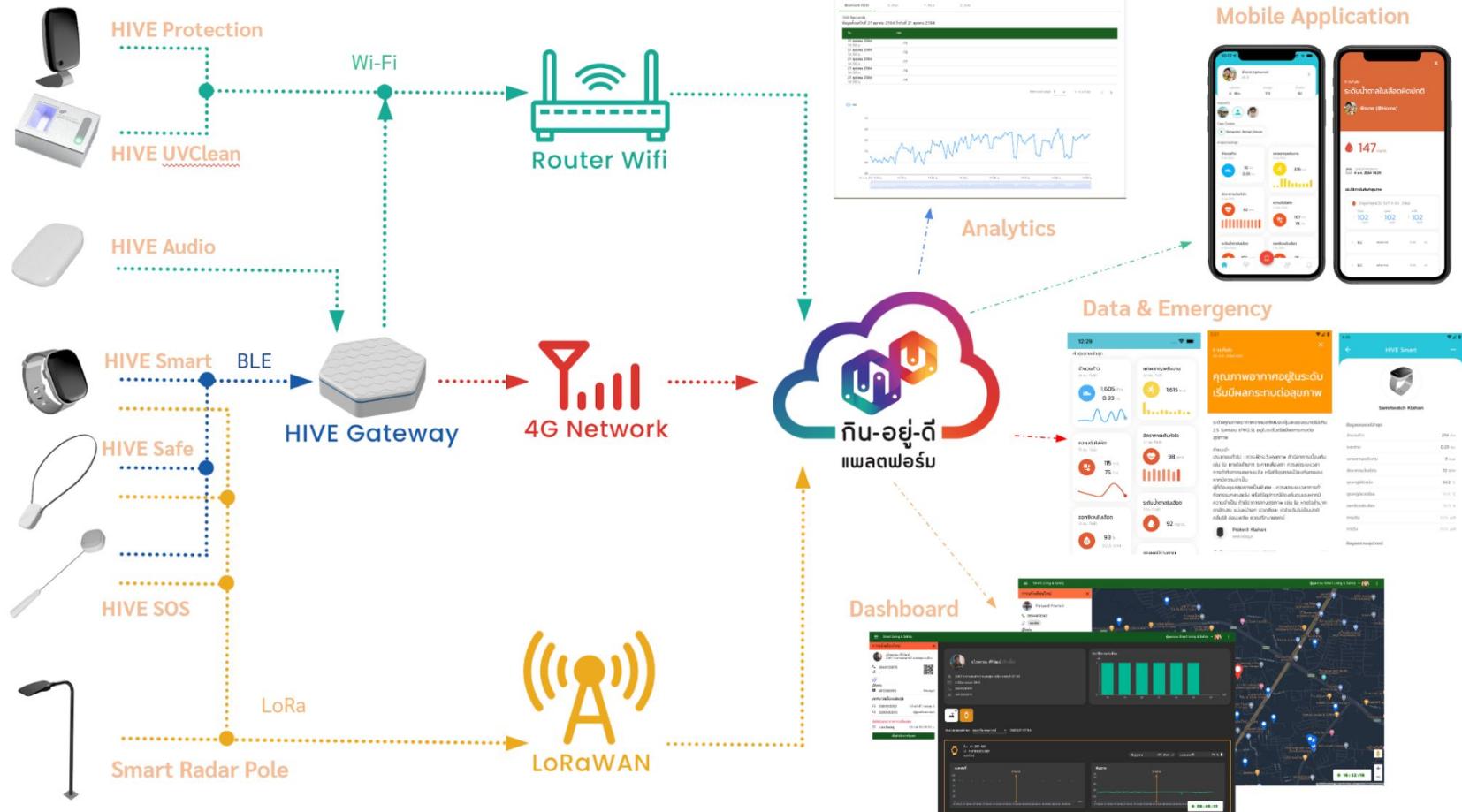


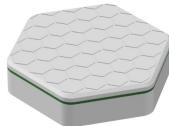
KIN-YOO-DEE PLATFORM

นวัตกรรมเพื่อยกระดับ
ความเป็นอยู่และสุขภาพที่ดี
อย่างยั่งยืน

ปัจจุบันบริษัทมีส่วนร่วมในการพัฒนาและให้บริการระบบและแอปพลิเคชัน
กิน-อยู่-ดี แพลตฟอร์มสำหรับแจ้งเตือนขอความช่วยเหลือฉุกเฉินสำหรับ
ผู้สูงอายุ การบันทึกกิจกรรมและข้อมูลสุขภาพเพื่อวิเคราะห์และประเมิน
สภาวะสุขภาพสมาชิกในครอบครัวและบริการชุมชน







HIVE Gateway

รับข้อมูลจากอุปกรณ์ IoT ด้วย Bluetooth และส่งข้อมูลไปยัง กิน-อยู่-ดี เพลตฟอร์มผ่านอินเตอร์เน็ต



HIVE Smart

นาฬิกาอัจฉริยะ สามารถตรวจจับการเคลื่อนไหว
บันทึกข้อมูลสุขภาพ ตรวจจับการล้ม ส่งสัญญาณขอความช่วยเหลือ



HIVE Safe

สร้อยคออัจฉริยะ สามารถตรวจจับการเคลื่อนไหว
ตรวจจับการล้ม ส่งสัญญาณขอความช่วยเหลือ



HIVE SOS

ปุ่มกดขอความช่วยเหลือในกรณีฉุกเฉิน



HIVE UV Clean

ฆ่าเชื้อโรคในอากาศด้วยรังสี UV



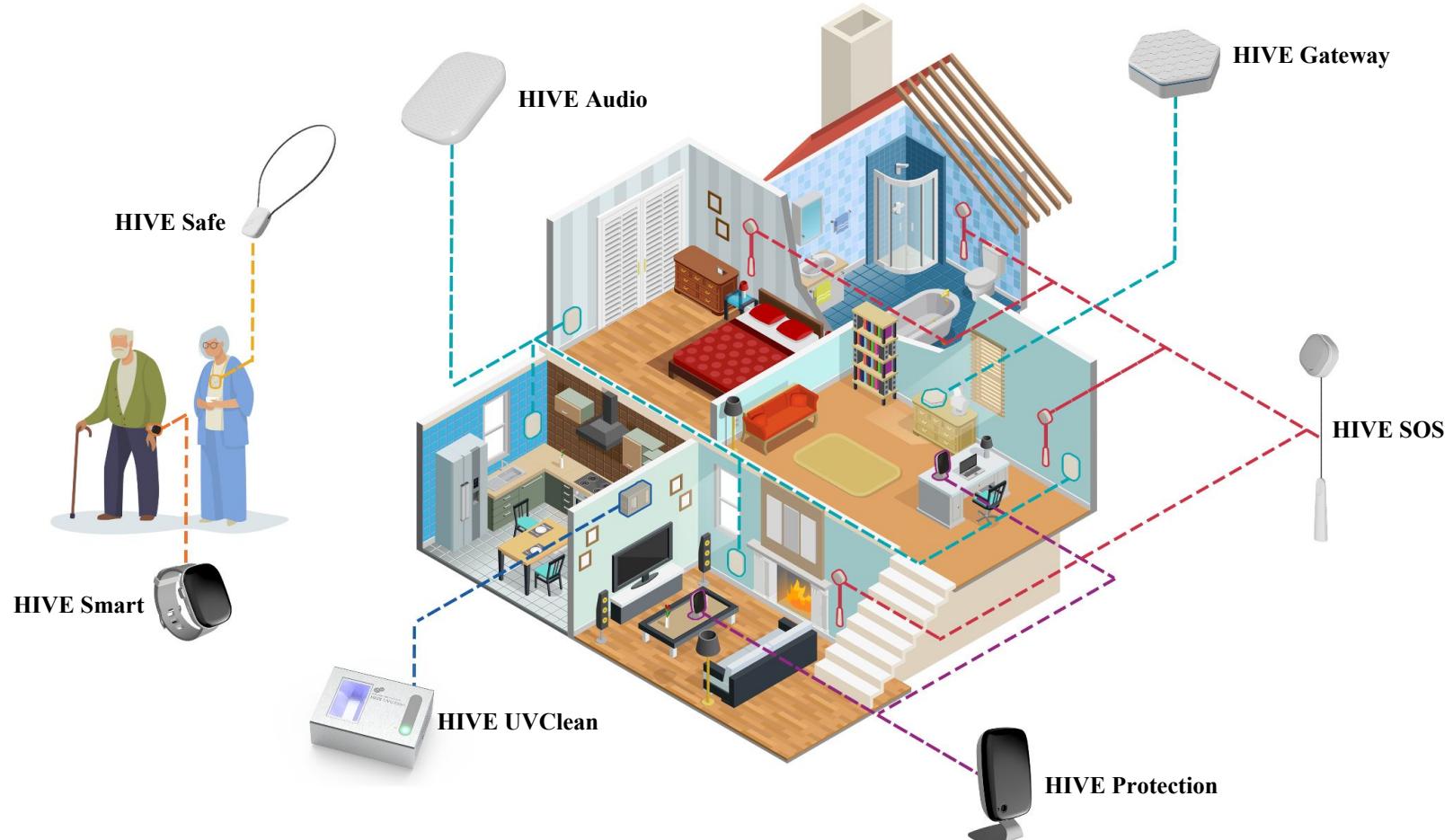
HIVE Protection Set

วัดค่าสภาวะอากาศ ค่า PM2.5 อุณหภูมิ ด้วย
เทคโนโลยี Radar



HIVE Audio

เชื่อมตือกับ Gateway และโทรด้วยเสียง



Agenda

IoT Devices

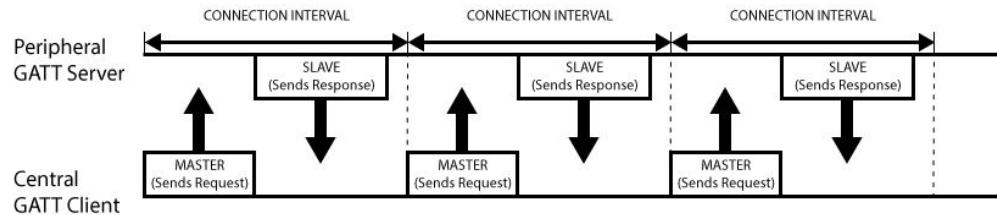
IoT Devices และ กิน-ອຍ-ดี แพลตฟอร์ม

1. Hexiwear
2. Bluetooth
3. Keil Studio Cloud
4. การส่งข้อมูลผ่าน Bluetooth
5. Q & A



Hexiwear คืออุปกรณ์ที่ใช้สำหรับพัฒนาเป็นอุปกรณ์ต้นแบบได้ เช่น Smartwatch, Remote sensor tag หรือตัวควบคุม SmartHome ซึ่งมีเซนเซอร์มาให้ภายในตัวสามารถเรียกใช้งานได้โดยไม่ต้องทำการเชื่อมต่อเพิ่ม เช่น Accelerometer Magnetometer Gyroscope เช่นเซอร์วัดอัตราการเต้นของหัวใจเซนเซอร์วัดความดันอากาศ เชอร์วัดความชื้น เช่นเซอร์วัดอุณหภูมิ และเซนเซอร์วัดแสงเป็นต้น ยังมีแบตเตอรี่ และตัวชาร์จแบตเตอรี่ขนาด 600mA ภายในตัว เมื่อต้องการขยายการใช้งานมากขึ้นสามารถเชื่อมต่อ Hexiwear เข้ากับบอร์ด Hexiwear Workstation เพื่อใช้งานกับอุปกรณ์ต่างๆ บนบอร์ดได้มากขึ้น

Bluetooth เป็นหนึ่งในเทคโนโลยีที่ใช้สื่อสารหรือรับส่งข้อมูลระหว่างอุปกรณ์สองตัวแบบไร้สาย โดยใช้คลื่นวิทยุความถี่ช่วง 2.4 GHz ใน การสื่อสาร เฉพเซ่นเดียวกัน WiFi เพียงแต่ว่าเป็นคนละโปรโตคอล (Protocol) กัน และมาตรฐาน Bluetooth อยู่ภายใต้การดูแลของ Bluetooth Special Interest Group (Bluetooth SIG) ที่ค่อยพัฒนาและกำหนดมาตรฐานของ Bluetooth ในแต่ละเวอร์ชัน



Peripheral (Slave) มักหมายถึงอุปกรณ์ขนาดเล็ก อาจเชื่อมต่อกับ Sensor จะค่อยส่งสัญญาณความต้องการ สื่อสารออกไป หรือ Broadcaster เมื่อการเชื่อมต่อ เรียบง่ายแล้วก็จะทำตามที่ Master กำหนด

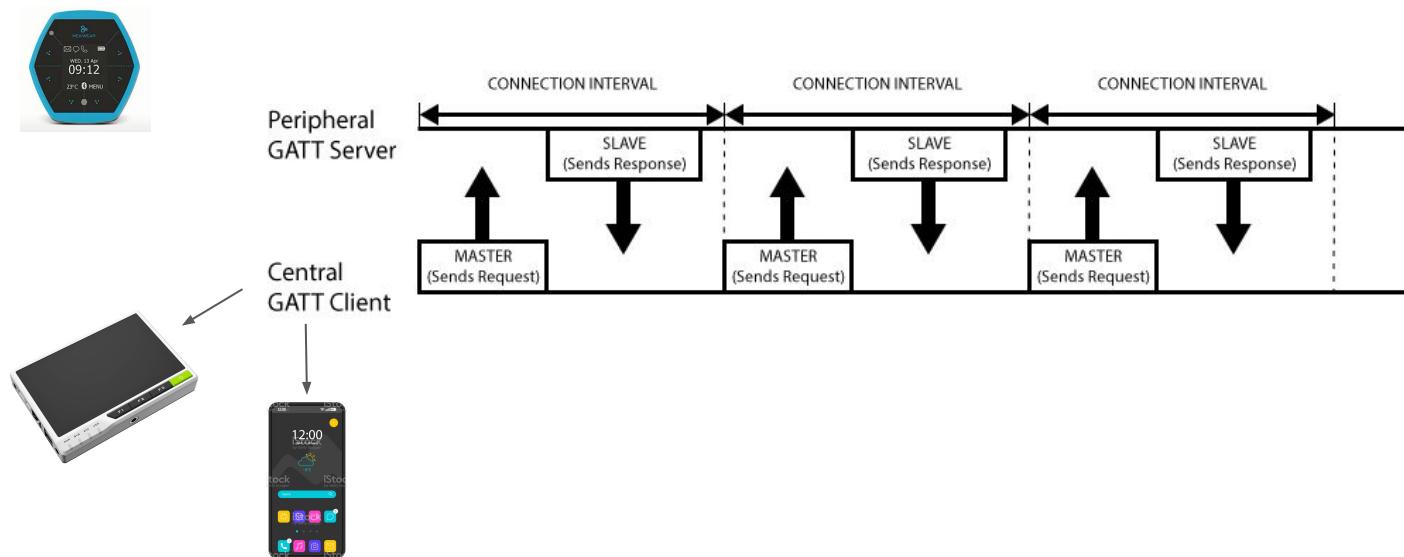


Central (Master) หมายถึงอุปกรณ์ที่มีความสามารถ สูง เช่น โทรศัพท์เคลื่อนที่ Tablet หรือ PC จะค่อยตรวจสอบ สัญญาณความต้องการการสื่อสารเป็นระยะ



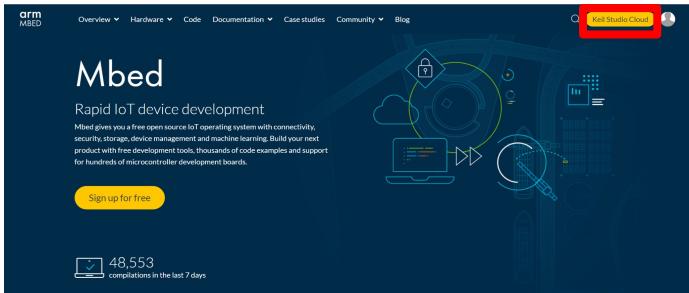
GAP (generic Access profile) คือ โครงสร้างโปรโตคอลบลูทูธที่ทำหน้าที่ควบคุมการเชื่อมต่อ การเข้าถึงอุปกรณ์ รวมถึงการค้นหาอุปกรณ์อุปกรณ์ด้วย เมื่ออุปกรณ์มีการเชื่อมต่อกันเรียบร้อยแล้วก็จะเป็นหน้าที่ของ

GATT (generic attribute profile) คือ โครงสร้างโปรโตคอลบลูทูธที่ทำหน้าที่ควบคุมการรับส่ง หรืออ่าน เขียนข้อมูลระหว่างตัว Peripheral (Slave) กับ Central (Master)



เข้าไปที่หน้าเว็บไซต์จากลิงค์ด้านล่าง และไปที่แท็บลูกศรด้านขวาบน และเลือกที่ Keil Studio Cloud

<https://os.mMbedbed.com/>



```
File Edit Selection View Go Help  
Active project Hexi_BLE_Example  
Target hardware Hexiwear  
main.cpp x  
1 #include "mexic.h"  
2 #include "Hexi_KW48Z.h"  
3 #include "Hexi_OLED_SSD1351.h"  
4 #include "OLED_types.h"  
5 #include "OpenSans_Font.h"  
6 #include "string.h"  
7  
8 #define LED_ON 0  
9 #define LED_OFF 1  
10  
11 void UpdateSensorsData(void);  
12 void StartHaptic(void);  
13 void StopHaptic(void const *);  
14 void txTask(void);  
15  
16 DigitalOut redled(LED0_1);  
17 DigitalOut greenled(LED0_2);  
18 DigitalOut blueled(LED0_3);  
19 DigitalOut haptic(PTE99);  
20  
21 // Define timer for haptic feedback /  
22 RtosTime hapticTime(stophaptic, osTimeOnce);  
23  
24 // Initialize the Hexi KW48Z Driver (UART TX, UART RX) /*  
KW48Z_kw48z_device(PTE24, PTE23);  
25  
26 // Instantiate the SSD1351 OLED Driver /*  
27 SSD1351 oled(PTB25, PTF21, PTB13, PTB20, PTE6, PTD15); /* (PSCI, SCLK, POWER_C5, RST, DC) */  
28  
Mbed Libraries x Output x  
Notifications x
```

<h1>Log in</h1> <p>New to Mbed? Create an account</p> <p>Email address * <input type="text"/></p> <p>Next</p> <p>Only know your username? Get an email reminder.</p> <hr/> <h2>Log in</h2> <p>New to Mbed? Create an account</p> <p>Email address * <input type="text"/></p> <p>Forgot your email?</p> <p>Password * <input type="password"/></p> <p>Forgot your password?</p> <p>Log in</p>	 <h1>Sign up</h1> <p>Already have an Mbed account? Log in</p> <p>Create a New Mbed account to access Mbed tools and services and contribute to the developer community.</p> <p>Email address *</p> <p>Username *</p> <p>Password *</p> <p>First name *</p> <p>Last name *</p> <p>Country *</p> <p>Thailand</p> <p>Arm will process your information in accordance with our Privacy Policy.</p> <p><input type="checkbox"/> By signing up, you indicate your consent to receive marketing communications from Arm in accordance with our Privacy Policy. Please visit our Subscription Center to manage your marketing preferences or unsubscribe from further communications.</p> <p><input type="checkbox"/> I agree to Arm's terms and conditions of use.*</p> <p>Sign up</p>
--	---

Log in

New to Mbed? [Sign up](#)

Email address *

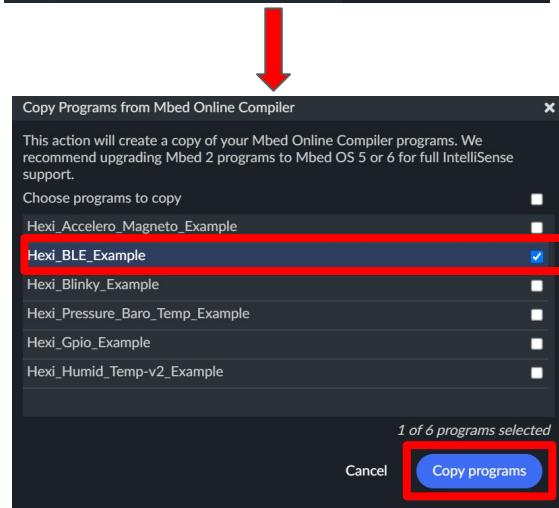
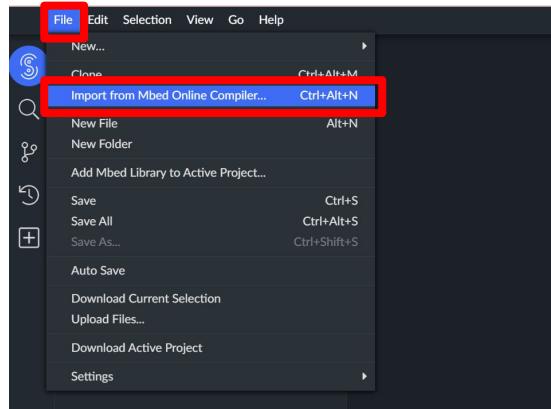
Forgotten your email?

Password *

Forgotten your password?

Log in

การส่งข้อมูลผ่าน Bluetooth



The code editor displays the 'main.cpp' file of the 'Hexi_BLE_Example' project. The code includes comments for displaying bond pass keys, registering callbacks, turning on the OLED display, and setting text properties. The 'main.cpp' tab is highlighted with a red box in the project tree. The code editor interface includes tabs for 'main.cpp', 'Mbed Libraries', and 'Output'.

```

/* Display Bond Pass Key in a 95px by 18px textbox at x=0,y=40 */
sprintf(text,"%d", kw40z_device.GetPassKey());
oled.TextBox((uint8_t *)text,0,40,95,18);

/* *****End of Call Back Functions***** */

/* *****Main***** */

int main()
{
    /* Register callbacks to application functions */
    kw40z_device.attach_buttonLeft(&ButtonLeft);
    kw40z_device.attach_buttonRight(&ButtonRight);
    kw40z_device.attach_passkey(&PassKey);

    /* Turn on the backlight of the OLED Display */
    oled.DimScreenOn();

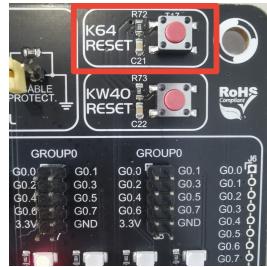
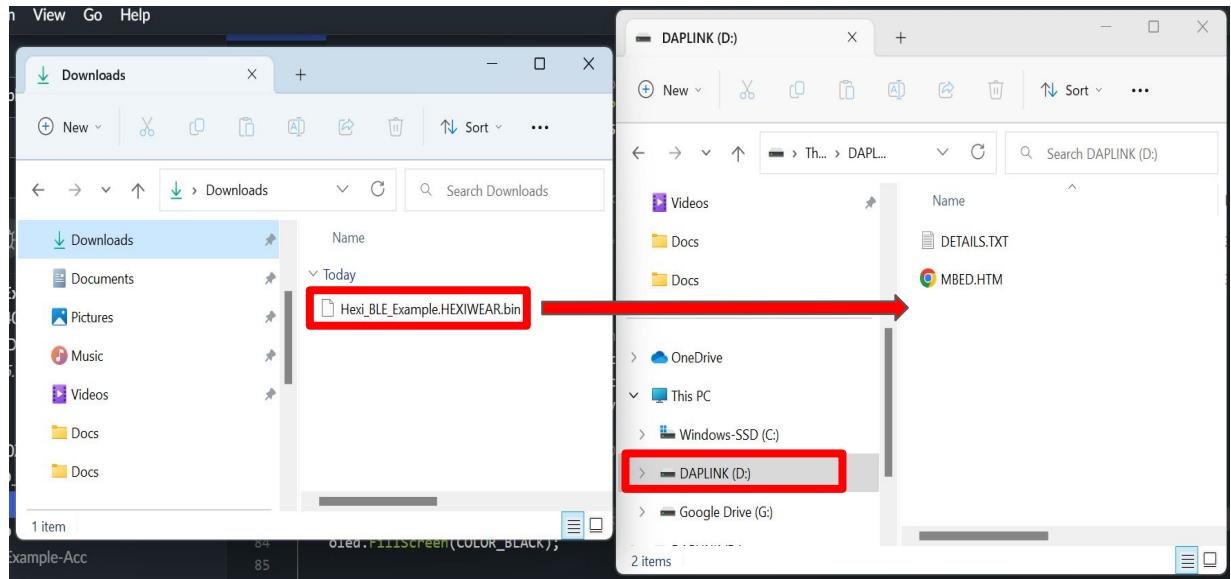
    /* Fills the screen with solid black */
    oled.FillScreen(COLOR_BLACK);

    /* Get OLED Class Default Text Properties */
    oled_text_properties_t textProperties = {0};
    oled.GetTextProperties(&textProperties);

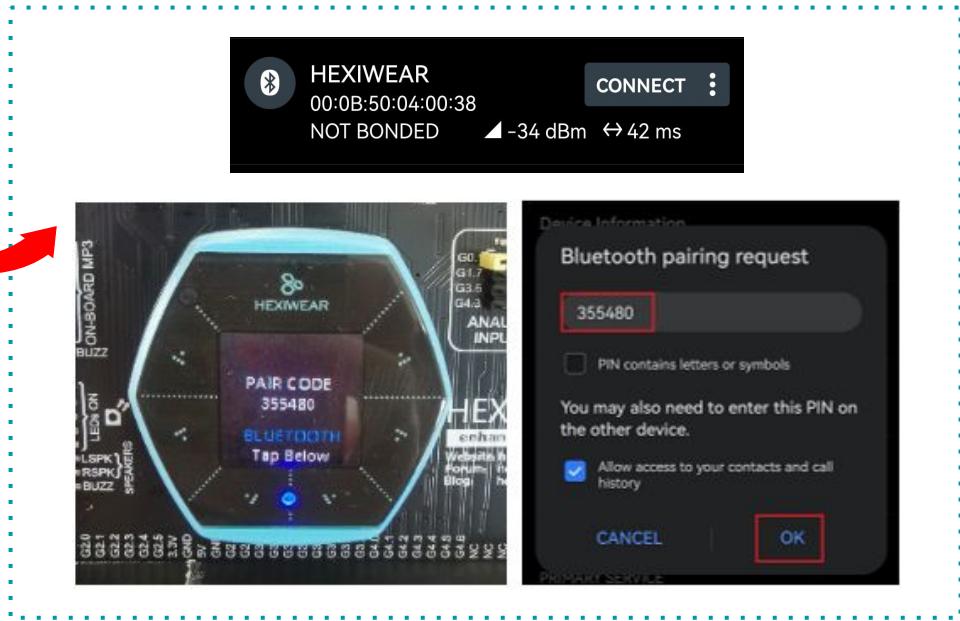
    /* Change font color to Blue */
    textProperties.textColor = COLOR_BLUE;
    oled.SetTextProperties(&textProperties);
}

```

การส่งข้อมูลผ่าน Bluetooth



การส่งข้อมูลผ่าน Bluetooth



Devices		DISCONNECT	⋮		
BONDED	ADVERTISER	HEXIWEAR 00:0B:50:04:00:38	X		
		CONNECTED BONDED	CLIENT	SERVER	⋮
Device Information					
UUID: 0x180A PRIMARY SERVICE					
Manufacturer Name String					
UUID: 0x2A29 Properties: READ Value: Mikroelektronika					
Firmware Revision String					
UUID: 0x2A26 Properties: READ Value: 1.0.2/1.0.0					
Serial Number String					
UUID: 0x2A25 Properties: READ Value: DM P B					
Battery Service					
UUID: 0x180F PRIMARY SERVICE					
Unknown Service					
UUID: 0x2000 PRIMARY SERVICE					
Unknown Service					
UUID: 0x2010 PRIMARY SERVICE					

การอ่านค่าจากเซนเซอร์การส่งข้อมูลผ่าน Bluetooth

ทำปรับโค้ด เพื่ออ่านค่าจากเซนเซอร์ Accelerometer

1. ทำการ Include FXOS8700.h

```
#include "FXOS8700.h"
```

2. ทำการสร้างตัวแปรสำหรับเก็บค่าที่อ่านได้จากเซนเซอร์ Accelerometer เป็นอาร์ยูขนาดเท่ากับ 3 ประเภท float และสร้างตัวแปรประเภท float อีกหนึ่งตัวเพื่อเก็บค่า rms ของเซนเซอร์ Accelerometer

```
float accel_data[3]; float accel_rms=0.0;
```

3. ทำการกำหนดพินสำหรับเชื่อมต่อเซนเซอร์ Accelerometer

```
FXOS8700 accel(PTC11, PTC10);
```

4. ทำการเรียกใช้ฟังก์ชัน Config เพื่อกำหนดค่าต่างๆ ให้กับเซนเซอร์ Accelerometer อู่ภายในฟังก์ชัน main()

```
// Configure Accelerometer FXOS8700
accel.accel_config();
```

5. ทำการเรียกใช้ฟังก์ชัน Config เพื่อกำหนดค่าต่างๆ ให้กับเซนเซอร์ Accelerometer อู่ภายในฟังก์ชัน main()

```
accel.acquire_accel_data_g(accel_data);
accel_rms =
sqrt((accel_data[0]*accel_data[0])+(accel_data[1]*accel_data[1])+(accel_data[2]*accel_data[2]))/3;
printf("Accelerometer \tX-Axis %4.2f \tY-Axis %4.2f \tZ-Axis %4.2f \tRMS
%4.2f\n\r",accel_data[0],accel_data[1],accel_data[2],accel_rms);
wait(0.01);
```

6. ทำการเขียนโปรแกรมสำหรับส่งข้อมูลค่าเซนเซอร์ Accelerometer ผ่าน Bluetooth โดยนำค่าที่อ่านได้จากเซนเซอร์มาคูณกับ 100 เพื่อจะส่งค่าออกไปเป็นจำนวนเต็ม เขียนโปรแกรมไว้ภายในลูป While ของฟังก์ชัน txTask(void) ดังนี้

```
kw40z_device.SendAccel((accel_data[0]*100),
(accel_data[1]*100),
(accel_data[2]*100)); // (z,x,y);
```

ดูค่าเซนเซอร์ Accelerometer จาก Serial Terminal และ



PuTTY

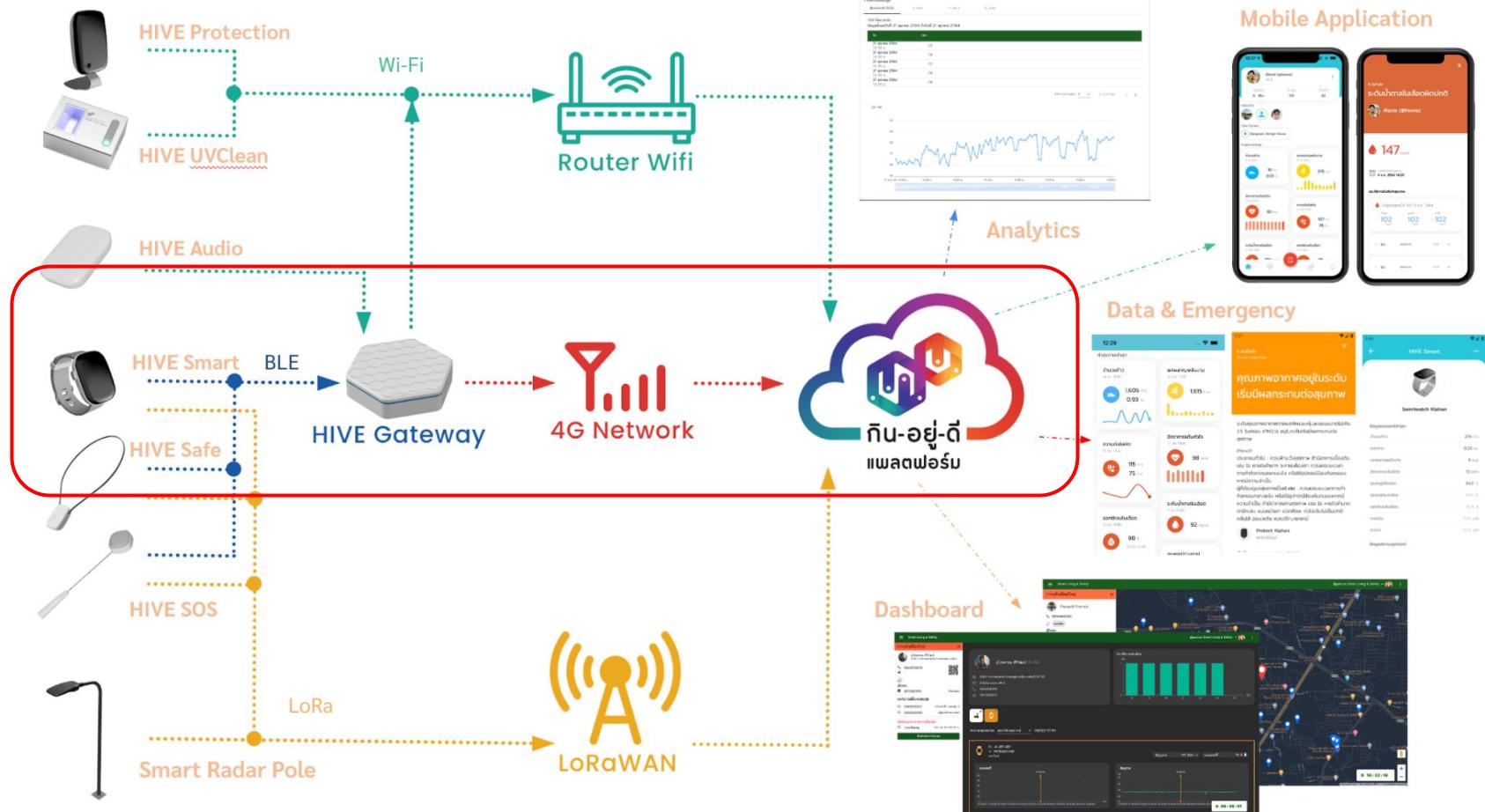
```
Batt_percent : 100%  
Accelerometer X-Axis 0.03      Y-Axis 0.00      Z-Axis -0.98    RMS 0.57  
Batt_percent : 100%  
Accelerometer X-Axis 0.04      Y-Axis -0.00     Z-Axis -0.98    RMS 0.57  
Batt_percent : 100%  
Accelerometer X-Axis 0.03      Y-Axis -0.01     Z-Axis -0.98    RMS 0.56  
Batt_percent : 100%  
Accelerometer X-Axis 0.03      Y-Axis -0.00     Z-Axis -0.98    RMS 0.57  
Batt_percent : 100%  
Accelerometer X-Axis 0.03      Y-Axis -0.00     Z-Axis -0.98    RMS 0.57
```



```
Unknown Service  
UUID: 0x2000  
PRIMARY SERVICE  
  
Unknown Characteristic  
UUID: 0x2001  
Properties: READ  
Value: (0x) FF-9F-00-03-00-00  
  
Unknown Characteristic  
UUID: 0x2002  
Properties: READ  
  
Unknown Characteristic  
UUID: 0x2003  
Properties: READ
```

IoT Devices และ กิน-อยู่-ดี แพลตฟอร์ม

1. IoT Network Connectivity
2. Raspberry Pi Gateway
3. การทดลองสร้าง IoT Gateway
 - เตรียม environment
 - Hexiware and Raspberry Connection
 - Read sensor value
 - Send sensor value to Platform
 - Monitoring and Visualization
4. Q & A

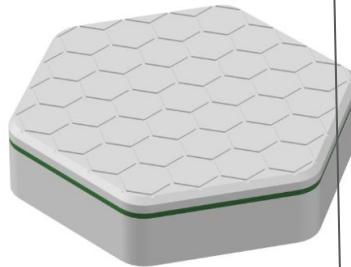




HIVE Smart



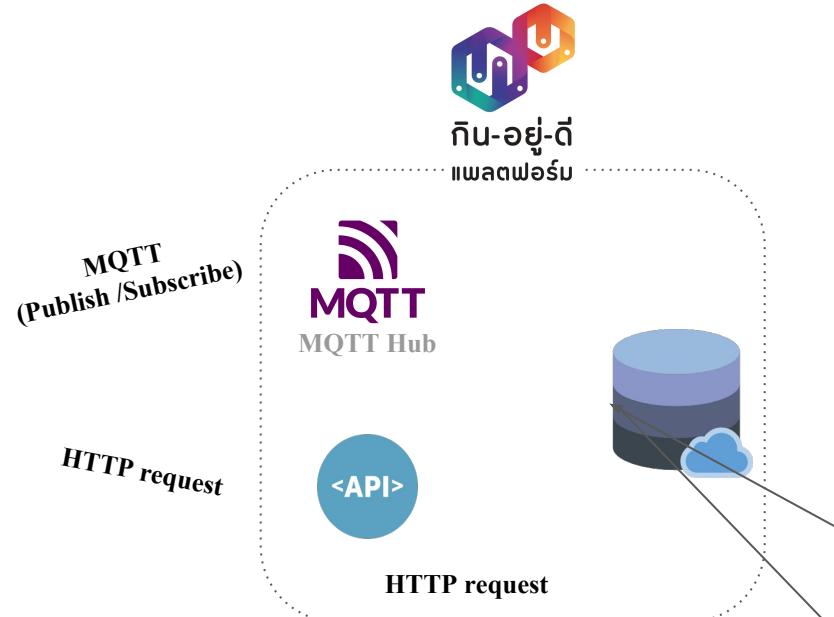
Hexiware



HIVE Gateway



reTerminal (Raspberry Pi)



```
pi@raspberrypi:~ $ bluetoothctl
Agent registered
[bluetooth]# scan on
Discovery started
[CHG] Controller E4:5F:01:2C:26:EC Discovering: yes
[NEW] Device 39:D6:F4:FB:91:55 39-D6-F4-FB-91-55
[NEW] Device 67:76:DE:9E:94:44 67-76-DE-9E-94-44
[CHG] Device 00:35:50:04:00:10 RSSI: -63
[NEW] Device 44:24:CF:58:1D:BC 44-24-CF-58-1D-BC
[NEW] Device 5A:C2:71:4E:9F:87 5A-C2-71-4E-9F-87
[NEW] Device 47:C4:47:05:C4:40 47-C4-47-05-C4-40
[NEW] Device 5F:D4:B2:C6:04:2E 5F-D4-B2-C6-04-2E
[NEW] Device 34:1F:AB:44:E2:F5 34-1F-AB-44-E2-F5
```

Bluetoothctl Using bluetooth with CLI tools

Example

```
sudo systemctl status bluetooth : check bluetooth status
```

bluetoothctl : using bluetoothctl

scan on - เริ่ม scan หาอุปกรณ์ลูกหลานฯ

scan off - หยุด scan

discoverable on - เพื่อสั่งให้อุปกรณ์อื่นๆ สามารถค้นหาอุปกรณ์ได้

discoverable off - เพื่อสั่งให้อุปกรณ์อื่นๆ ไม่สามารถค้นหาอุปกรณ์ได้

pair XX:XX:XX:XX:XX:XX - เพื่อสั่งให้จับคู่อุปกรณ์ตาม Mac Address

remove XX:XX:XX:XX:XX:XX - เพื่อสั่งให้ยกเลิกจับคู่อุปกรณ์ตาม Mac Address

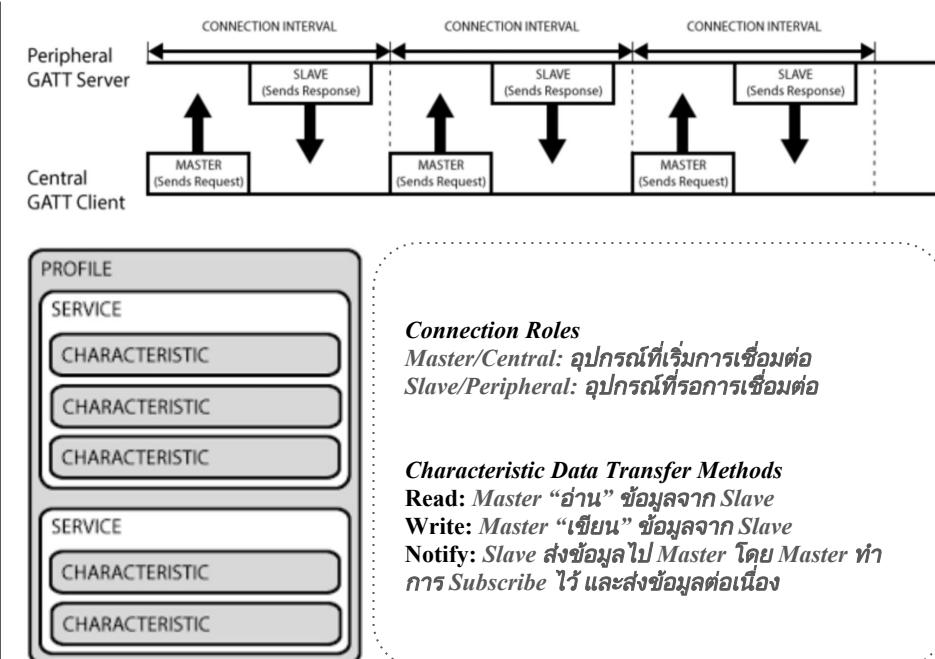
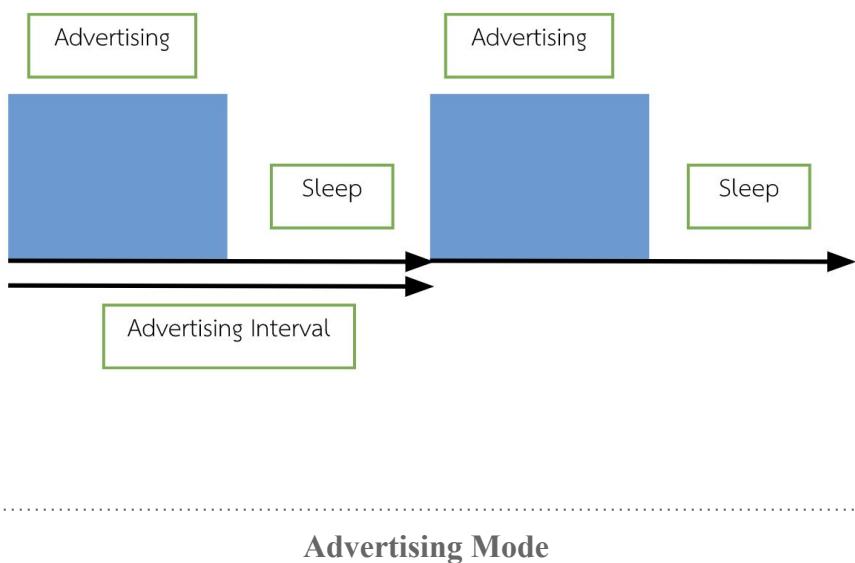
connect XX:XX:XX:XX:XX:XX - เพื่อสั่งให้เชื่อมต่ออุปกรณ์ตาม Mac Address

disconnect XX:XX:XX:XX:XX:XX - เพื่อสั่งให้ยกเลิกการเชื่อมต่ออุปกรณ์ตาม Mac Address



Bluetooth Low Energy: BLE -> Bluetooth v4.x

Mode: Advertising / Connect





Javascript: ใช้หน้าเว็บ Run ผ่าน Web Browser

nodejs: เป็น interpreter / Runtime ให้ javascript ทำงานได้โดยไม่ต้องมี Web Browser

npm: ตัวจัดการ package สำหรับใช้งานร่วมกับ package จากที่อื่นๆ

```
pi@raspberrypi: ~$ ls
pi@raspberrypi: ~$ node -v
pi@raspberrypi: ~$ wget https://nodejs.org/dist/v8.9.0/node-v8.9.0-linux-armv7l.tar.gz
pi@raspberrypi: ~$ tar -xvf node-v8.9.0-linux-armv7l.tar.gz
pi@raspberrypi: ~$ cd node-v8.9.0-linux-armv7l/
pi@raspberrypi: ~/node-v8.9.0-linux-armv7l $ sudo cp -R * /usr/local/
pi@raspberrypi: ~/node-v8.9.0-linux-armv7l $ 
```



noble: nodejs package สำหรับใช้งานร่วมกับ ble (Bluetooth Low Energy)

Installation: npm install noble

Usage: var noble = require('noble');

```
pi@raspberrypi: ~$ ssh
pi@raspberrypi: ~$ noble
Connected to peripheral: 003550040010
Sending success !
Sending success !
Sending Data ...
Sending success !
Accelerometer Data >>> X:-0.97, Y:0.01, Z:0
Battery Data >>> 10%
Sending success !
Reading Data ...
Accelerometer Data >>> X:-0.98, Y:0.02, Z:0
Battery Data >>> 85%
Sending success !
Reading Data ...
Accelerometer Data >>> X:-0.98, Y:0.02, Z:0
Battery Data >>> 65%
```

```
Last login: Mon Jan 23 00:22:27 on ttys003
tuokon@Knuawes MacBook-Pro ~ ssh pi@192.168.1.153
pi@192.168.1.153:~$ password:
Linux raspberrypi 5.15.76-v7l #1597 SMP Fri Nov 4 12:14:58 GMT 2022 armv7l

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Mon Jan 23 00:22:47 2023 from 192.168.1.102

SSH is enabled and the default password for the 'pi' user has not been changed.
This is a security risk - please login as the 'pi' user and type 'passwd' to set
a new password.

pi@raspberrypi:~$ bluetoothctl
Agent registered
[00:35:50:04:00:10] Connected: yes
[HEXWEAR]# 
```

MQTT

1. Broker / Client
2. Topic / Wildcard
3. Publish / Subscribe

myhome / ground_floor / living_room / temp

myhome / ground_floor / + / temp

myhome / ground_floor / #



Client

Publish (to broker)



Subscribe (Wait for topic change)



Client

Subscribe (Wait for topic change)



Broker



Client

Subscribe (Wait for topic change)

Publish (to broker)



Client

ការអភិវឌ្ឍន៍ IoT Gateway > ពេទ្យលេខកូដ > MQTT



A
MQTT Client

PUB: myhome / ground_floor / living_room / temp
{temp : '28.5 °C'}



B
MQTT Client

PUB: myhome / ground_floor / kitchen / temp
{temp : '32.1 °C'}



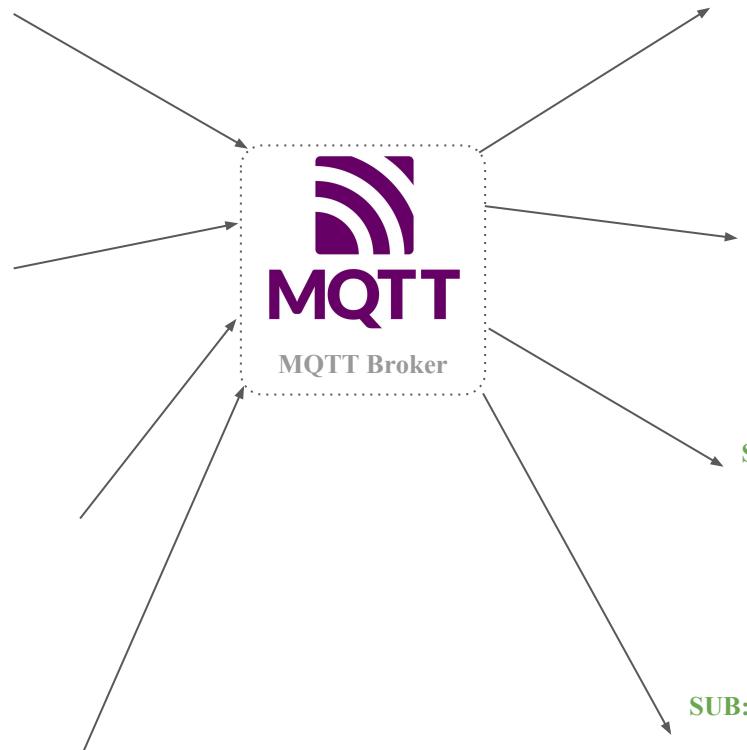
C
MQTT Client

PUB: myhome / ground_floor / living_room / brightness
{light : '1300'}



D
MQTT Client

PUB: myhome / second_floor / bed_room / temp
{temp : '28.5 °C'}



1
MQTT Client

SUB: myhome / ground_floor / + / temp



2
MQTT Client

SUB: myhome / ground_floor / #



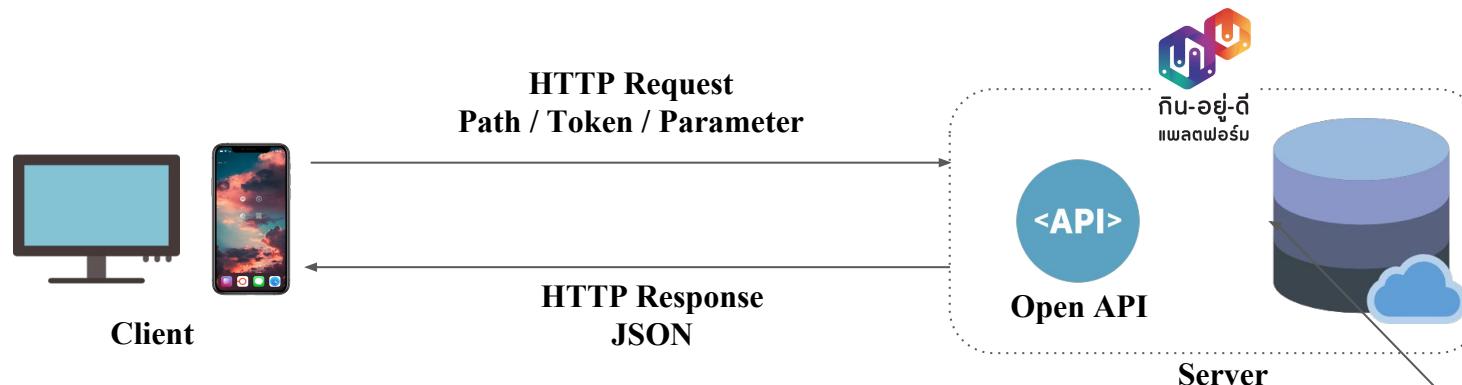
3
MQTT Client

SUB: myhome / second_floor / #



4
MQTT Client

SUB: myhome / ground_floor / living_room / #



Devices STOP SCANNING :

SCANNER BONDED ADVERTISER

No filter
Apple, Microsoft, Samsung, Google, Exposure Notification Service

HEXIWEAR 00:35:50:04:00:10 NOT BONDED		
	▲ -45 dBm	↔ 49 ms
N/A 4C:FB:F4:70:AD:F6 NOT BONDED		
	▲ -68 dBm	↔ 270 ms
N/A E4:75:2D:54:E6:1E NOT BONDED		
	▲ -53 dBm	↔ 1287 ms

```
pi@raspberrypi:~ $ bluetoothctl
Agent registered
[bluetooth]# scan on
Discovery started
[CHG] Controller E4:5F:01:2C:26:EC Discovering: yes
[NEW] Device 39:D6:F4:FB:91:55 39-D6-F4-FB-91-55
[NEW] Device 67:76:DE:9E:94:44 67-76-DE-9E-94-44
[CHG] Device 00:35:50:04:00:10 RSSI: -63
[NEW] Device 44:24:CF:58:1D:BC 44-24-CF-58-1D-BC
[NEW] Device 5A:C2:71:4E:9F:87 5A-C2-71-4E-9F-87
[NEW] Device 47:C4:47:05:C4:40 47-C4-47-05-C4-40
[NEW] Device 5F:D4:B2:C6:04:2E 5F-D4-B2-C6-04-2E
[NEW] Device 34:1F:AB:44:E2:F5 34-1F-AB-44-E2-F5
```



```
[bluetooth]# pair 00:35:50:04:00:10
Attempting to pair with 00:35:50:04:00:10
[CHG] Device 00:35:50:04:00:10 Connected: yes
Request passkey
[agent] Enter passkey (number in 0-999999): [CHG] Device 00:35:50:04:00:10 ServicesResolved: yes
231318
```

```
[CHG] Device 00:35:50:04:00:10 Paired: yes
Pairing successful
```

Read sensor value

```

noble.on("discover", async function (peripheral) {
  if (peripheral.address === DEVICE_MAC_ADDRESS) {
    stopScan();
    peripheral.connect(async function (error) {
      const serviceUUIDs = [BATTERY_SERVICE_UUID, SENSOR_SERVICE_UUID];
      const characteristicUUIDs = [
        BATTERY_CHARACTERISTIC_UUID,
        ACCELEROMETER_CHARACTERISTIC_UUID,
      ];
      peripheral.discoverSomeServicesAndCharacteristics(
        serviceUUIDs,
        characteristicUUIDs,
        (error, services, characteristics) => {
          console.log("Connected to peripheral: " + peripheral.uuid);

          const ACCELEROMETER_CHARACTERISTIC = characteristics.find(
            (char) => char.uuid === ACCELEROMETER_CHARACTERISTIC_UUID
          );

          const BATTERY_CHARACTERISTIC = characteristics.find(
            (char) => char.uuid === BATTERY_CHARACTERISTIC_UUID
          );

          let accel_x = undefined,
            accel_y = undefined,
            accel_z = undefined,
            batt = -1;

          setInterval(() => {
            console.log("Reading Data ...");
            ACCELEROMETER_CHARACTERISTIC.read((error, data) => {
              [accel_x, accel_y, accel_z] = convertAccelerometerData(data);
              console.log(
                "Accelerometer Data >>> ",
                `X:${accel_x}, Y:${accel_y}, Z:${accel_z}`
              );
            });
            BATTERY_CHARACTERISTIC.read((error, data) => {
              batt = data[0];
              console.log("Battery Data >>> ", `${batt}%`);
            });
          }, 5000);
        }
      );
    });
  }
});

```

```

pi@raspberrypi:~/hexiwear-connect $ sudo node index.js
Scan started
Scan stopped
Connected to peripheral: 003550040010
Reading Data ...
Accelerometer Data >>> X:-0.98, Y:0.01, Z:0
Battery Data >>> 25%

```

ลงทะเบียนอุปกรณ์

จำนวน - 25 ออนไลน์ - 0 ออฟไลน์ - 25

ค้นหา	องค์กร	ลงทะเบียนอุปกรณ์					
ความปลอดภัย	ชื่ออุปกรณ์	รหัสอุปกรณ์	ประเภทอุปกรณ์	เข้าออก	สถานะ	ออนไลน์	操作
<input checked="" type="checkbox"/>	Hexiwear2	00295004000E	Training_Board	-	กำลังใช้งาน	●	
<input checked="" type="checkbox"/>	Hexiwear	003550040010	Training_Board	-	กำลังใช้งาน	●	
<input checked="" type="checkbox"/>	HA [1]	TNB000500	Training_Board	-	กำลังใช้งาน	●	
<input checked="" type="checkbox"/>	Hexiwear	000B50040038	Training_Board	-	กำลังใช้งาน	●	
<input checked="" type="checkbox"/>	Gas Sensor [1]	TNB000504	Training_Board	-	กำลังใช้งาน	●	
<input checked="" type="checkbox"/>	Motion Sensor [1]	TNB000503	Training_Board	-	กำลังใช้งาน	●	
<input checked="" type="checkbox"/>	Door Sensor [1]	TNB000502	Training_Board	-	กำลังใช้งาน	●	
<input checked="" type="checkbox"/>	Power Meter [1]	TNB000501	Training_Board	-	กำลังใช้งาน	●	
<input checked="" type="checkbox"/>	13DOF Click [1]	TNB000023	Training_Board	-	กำลังใช้งาน	●	
<input checked="" type="checkbox"/>	Earth Quake Click [1]	TNB000022	Training_Board	-	กำลังใช้งาน	●	

Items per page:

10

1 - 10 of 25



Send sensor value to Platform > ส่งข้อมูลผ่าน Open API

```

function sendData(macAddress, accel_x, accel_y, accel_z, batt) {
  /* import module */ - Untitled-1
    id: macAddress.toUpperCase().replace(/:/g, ""),
    type: 380001,
    data: {
      dts: new Date().getTime(),
      seq: 1,
      status: 1,
      fwVersion: "1.0.0",
      hexiwear: {
        accel_x: accel_x,
        accel_y: accel_y,
        accel_z: accel_z,
        batt: batt,
      },
    },
  };
  data = JSON.stringify(payload);
  const options = {
    hostname: "api.aidery.io",
    path: "/data-log",
    method: "POST",
    headers: {
      Authorization: `Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpZCI6IjYyOGNhZWlkMTgxODU5MDAx
      "Content-Type": "application/json",
      "Content-Length": data.length,
    },
  };
  const req = https
    .request(options, (res) => {
      let data = "";
      res.on("data", (chunk) => {
        data += chunk;
      });
      res.on("end", () => {
        console.log("Sending success !");
      });
    })
    .on("error", (err) => {
      console.log("HTTP Request Fail !");
    });

  req.write(data);
  req.end();
}

```

Monitoring and Visualization > Realtime data logging

Training Board

ห้องเรียนอุปกรณ์

จำนวน - 24 ออนไลน์ - 7 ไม่ออนไลน์ - 17

ลำดับ	ชื่ออุปกรณ์	รหัสอุปกรณ์	ประเภทอุปกรณ์	เข้าออก	สถานะ	操作
1	Hexiwear	003550040010	Training_Board	-	กำลังใช้งาน	
2	HA [1]	TNB000500	Unknown_Type	-	กำลังใช้งาน	
3	Hexiwear	000850040038	Unknown_Type	-	กำลังใช้งาน	
4	Gas Sensor [1]	TNB000504	Unknown_Type	-	กำลังใช้งาน	
5	Motion Sensor [1]	TNB000503	Unknown_Type	-	กำลังใช้งาน	
6	Door Sensor [1]	TNB000502	Unknown_Type	-	กำลังใช้งาน	
7	Power Meter [1]	TNB000501	Unknown_Type	-	กำลังใช้งาน	
8	13DOF Click [1]	TNB000023	Unknown_Type	-	กำลังใช้งาน	
9	Earth Quake Click [1]	TNB000022	Unknown_Type	-	กำลังใช้งาน	
10	IR Gesture Click [1]	TNB000021	Unknown_Type	-	กำลังใช้งาน	

Items per page:

Hexiwear

Device Information

Mac Address: 003550040010

Model: TNB Model A

Serial No.: TNB-003550040010

Log Details

From: 21 มี. 65 16:53 น.

To: 23 มี. 65 17:38 น.

Total Record: 85 Records

Logging Controller

Realtime:

Interval: 5 sec

```
Type 380001 - Timestamp 2022-06-23T10:38:37.698Z
{
  "dts": 1655980717281,
  "seq": 1,
  "status": 1,
  "fwVersion": "1.0.0",
  "hexiwear": {
    "accel_x": -0.96,
    "accel_y": 0.02,
    "accel_z": 0,
    "batt": -1
  }
}
=====
Type 380001 - Timestamp 2022-06-23T10:38:42.669Z
{
  "dts": 1655980722291,
  "seq": 1,
  "status": 1,
  "fwVersion": "1.0.0",
  "hexiwear": {
    "accel_x": -0.96,
    "accel_y": 0.02,
    "accel_z": 0,
    "batt": 50
  }
}
=====
Type 380001 - Timestamp 2022-06-23T10:38:47.573Z
{
  "dts": 165598072286,
  "seq": 1,
  "status": 1,
  "fwVersion": "1.0.0",
  "hexiwear": {
    "accel_x": -0.96,
    "accel_y": 0.02,
    "accel_z": 0,
    "batt": 50
  }
}
```

Monitoring and Visualization > Data visualization (Chart)

5 Record

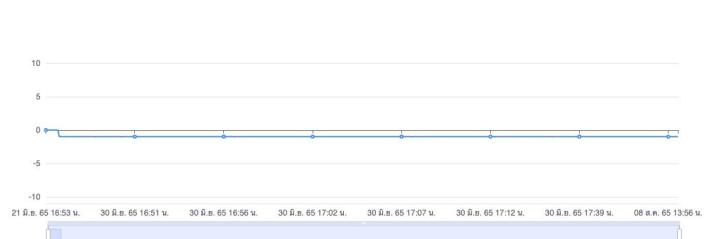
Device Offline Time



2319 Records
จำนวนทั้งหมดที่ 21 มิถุนายน 2565 ถึงวันที่ 08 สิงหาคม 2565

วัน	เวลา	สถานะ
08 สิงหาคม 2565	13:54 น.	-0.97
08 สิงหาคม 2565	13:56 น.	-0.97
08 สิงหาคม 2565	13:56 น.	-0.97
08 สิงหาคม 2565	13:56 น.	-0.97
08 สิงหาคม 2565	13:56 น.	-0.97
08 สิงหาคม 2565	13:56 น.	-0.97

Items per page: 5 ▾ 1 - 5 of 2319 < >



Any Question ?

1. IoT Network Connectivity - KYD's Platform
2. Hexiware and Raspberry Pi Gateway
3. การทดลองสร้าง IoT Gateway
 - เตรียม environment
 - Hexiware and Raspberry Connection
 - Read sensor value
 - Send sensor value to Platform
 - Monitoring and Visualization
4. Q & A