**電通二乙微處理器實驗 實驗結報**

|  |  |  |  |
| --- | --- | --- | --- |
| **實驗名稱** | **Lab 09-Bluetooth low energy** | | |
| **組別** |  | **組員** | **吳東燁** |

1. **實驗目的**
2. **實驗步驟**

**由手機 App 看到自己的 7697 iBeacon**

**由手機 App 看到自己的 7697 EddyStone URL Beacon**

**由手機 App 控制 7697 LED**

1. **程式碼**

|  |
| --- |
| **1.** |
| **/\***  **This example configures LinkIt 7697 to send iBeacon-compatbile advertisement data.**  **You should be able to search this device with iOS or Android iBeacon tools.**  **created Mar 2017**  **\*/**  **#include <LBLE.h>**  **#include <LBLEPeriphral.h>**  **void setup() {**  **//Initialize serial and wait for port to open:**  **Serial.begin(9600);**  **// Initialize BLE subsystem**  **Serial.println("BLE begin");**  **LBLE.begin();**  **while (!LBLE.ready()) {**  **delay(100);**  **}**  **Serial.println("BLE ready");**  **// configure our advertisement data as iBeacon.**  **LBLEAdvertisementData beaconData;**  **// This is a common AirLocate example UUID.**  **LBLEUuid uuid("06050934-DFFB-48D2-B060-D0F5A71096E0");**  **beaconData.configAsIBeacon(uuid, 01, 02, -40);**  **Serial.print("Start advertising iBeacon with uuid=");**  **Serial.println(uuid);**  **// start advertising it**  **LBLEPeripheral.advertise(beaconData);**  **}**  **void loop() {**  **// The underlying framework will advertise periodically.**  **// we simply wait here.**  **//**  **// You can use iBeacon apps such as**  **// "Locate Beacon" by Radius Networks on iOS devices**  **// to locate this beacon.**  **delay(3000);**  **}** |

|  |
| --- |
| **2.** |
| **/\***  **This example configures LinkIt 7697 to send Eddyston-URL advertisement data.**  **You should be able to search this beacon with tools such as "Beacon Tools" on iOS or**  **"Physical Web" app on Android.**  **created April 2017**  **\*/**  **#include <LBLE.h>**  **#include <LBLEPeriphral.h>**  **void setup() {**  **//Initialize serial and wait for port to open:**  **Serial.begin(115200);**  **// Initialize BLE subsystem**  **Serial.println("BLE begin");**  **LBLE.begin();**  **while (!LBLE.ready()) {**  **delay(100);**  **}**  **Serial.println("BLE ready");**  **// configure our advertisement data as iBeacon.**  **LBLEAdvertisementData beaconData;**  **// make an Eddystone-URL beacon that board casts**  **// https://labs.mediatek.com**  **// Note 1: You can obmit the suffix and tail part, e.g.**  **// https://goo.gl/Aq18zF**  **// can be constructed with**  **// configAsEddystoneURL(EDDY\_HTTPS, "goo.gl/Aq18zF");**  **// Note 2: Note that total url length must not exceed 17 bytes.**  **//**  **// Please refer to https://github.com/google/eddystone/tree/master/eddystone-url#url-scheme-prefix**  **// to know how the prefix/suffix/tails are expanded.**  **beaconData.configAsEddystoneURL(EDDY\_HTTPS, "06050934", EDDY\_DOT\_COM);**  **Serial.print("Start advertising Eddystone-URL");**  **// start advertising it**  **LBLEPeripheral.advertiseAsBeacon(beaconData);**  **}**  **void loop() {**  **// The underlying framework will advertise periodically.**  **// we simply wait here.**  **//**  **// You should be able to search this beacon with tools such as "Beacon Tools" on iOS or**  **// "Physical Web" app on Android.**  **delay(3000);**  **}** |

|  |
| --- |
| **3.** |
| **/\***  **This example configures LinkIt 7697 to act as a simple GATT server with 1 characteristic.**  **To use it, open AppInventor project:**  **\***  **Build & install it on Android id**  **created Mar 2017**  **\*/**  **#include <LBLE.h>**  **#include <LBLEPeriphral.h>**  **// Define a simple GATT service with only 1 characteristic**  **LBLEService ledService("06050934-E8F2-537E-4F6C-D104768A1214");**  **LBLECharacteristicInt switchCharacteristic("19B10011-E8F2-537E-4F6C-D104768A1214", LBLE\_READ | LBLE\_WRITE);**  **void setup() {**  **// Initialize LED pin**  **pinMode(LED\_BUILTIN, OUTPUT);**  **digitalWrite(LED\_BUILTIN, LOW);**  **//Initialize serial and wait for port to open:**  **Serial.begin(9600);**  **// to check if USR button is pressed**  **pinMode(6, INPUT);**  **// Initialize BLE subsystem**  **LBLE.begin();**  **while (!LBLE.ready()) {**  **delay(100);**  **}**  **Serial.println("BLE ready");**  **Serial.print("Device Address = [");**  **Serial.print(LBLE.getDeviceAddress());**  **Serial.println("]");**  **// configure our advertisement data.**  **// In this case, we simply create an advertisement that represents an**  **// connectable device with a device name**  **LBLEAdvertisementData advertisement;**  **advertisement.configAsConnectableDevice("Liann");**  **// Configure our device's Generic Access Profile's device name**  **// Ususally this is the same as the name in the advertisement data.**  **LBLEPeripheral.setName("Liann");**  **// Add characteristics into ledService**  **ledService.addAttribute(switchCharacteristic);**  **// Add service to GATT server (peripheral)**  **LBLEPeripheral.addService(ledService);**  **// start the GATT server - it is now**  **// available to connect**  **LBLEPeripheral.begin();**  **// start advertisment**  **LBLEPeripheral.advertise(advertisement);**  **}**  **void loop() {**  **delay(1000);**  **Serial.print("conected=");**  **Serial.println(LBLEPeripheral.connected());**  **if (digitalRead(6))**  **{**  **Serial.println("disconnect all!");**  **LBLEPeripheral.disconnectAll();**  **}**  **if (switchCharacteristic.isWritten()) {**  **const char value = switchCharacteristic.getValue();**  **switch (value) {**  **case 1:**  **digitalWrite(LED\_BUILTIN, HIGH);**  **break;**  **case 0:**  **digitalWrite(LED\_BUILTIN, LOW);**  **break;**  **default:**  **Serial.println("Unknown value written");**  **break;**  **}**  **}**  **}** |

1. **實驗結果及分析**
2. **心得討論**

**這周讓我體驗藍芽的使用方法，不錯**