

ES_Lab2_report

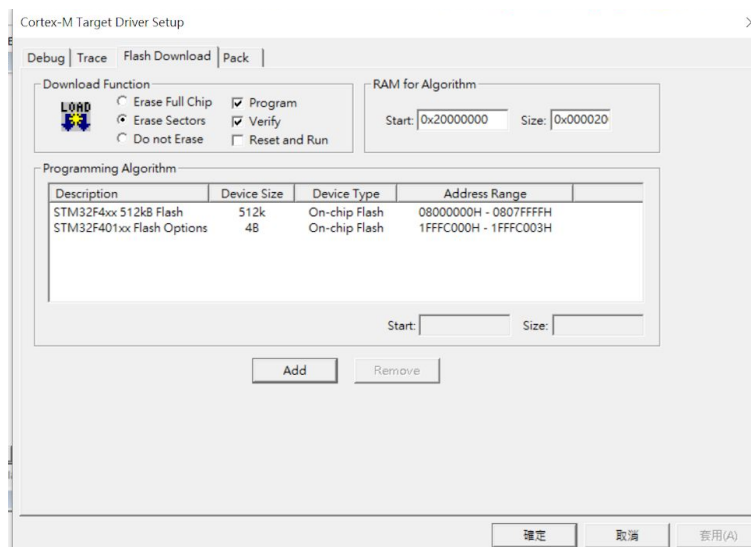
組員：B06502141丁冠廷、B06901045曹林熹、B06901188李宗倫

Before Lab2

在我們做Lab2的三個引導實驗時，首先在硬體燒入程式，及軟體的上面我們碰到了以下三大問題，這裡一一說明我們的解法：

1. flash

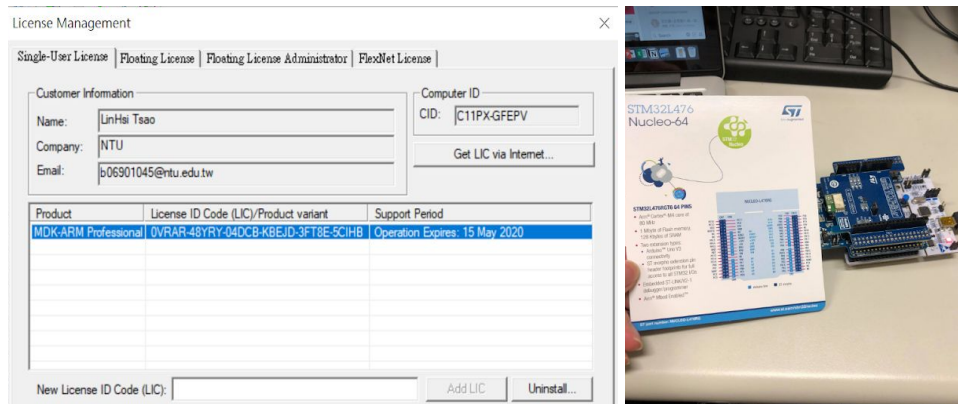
這裡遇到的第一個問題為將code compile完後，我們要將程式硬load到硬體時，會跳出一個Flash Download failed - Target DLL has been cancelled，推測是硬體本身自己的flash容量不夠，導致程式燒入時被拒絕，因此我們上網去找相關的Debug方法，最後找到要去option的設定加入STM32F401 xx Flash Options這個選項，而更改設定後有順利解決這個error。



2. license 記得用管理員身分 add

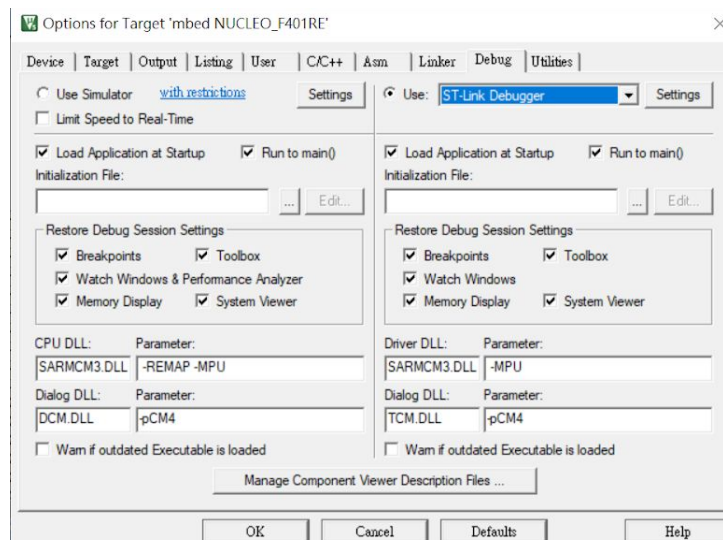
這裡遇到的第二個error一樣是出現在load程式進入板子時，跳出Error：Flash download failed: could not load file ...\\Module\\Module11，後來上網找尋將這個問題解決的方法，發現是硬體本身無法承載程式大小的問題，解法之一是換更新的板子，而因為我們手邊的資源沒辦法以這個方法解決，而找到另一種方法：將keil的license更新。

這個方法我們也覺得蠻神奇的，為何更新license就能解決硬體問題，算是Lab2 裡我們蠻大的一個疑問，以下是更新後的情形。



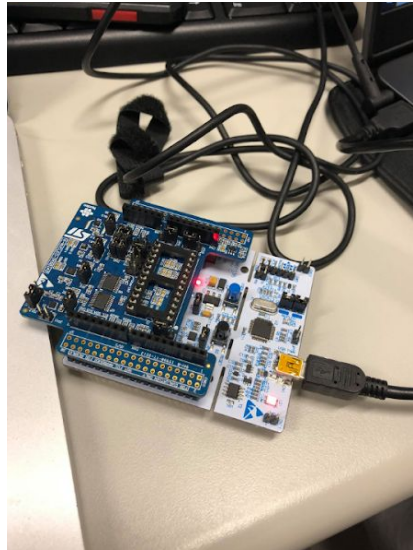
3. debugger setting

第三個在執行程式時我們遇到的error，為Debugger的設定問題，我們推測是不同的task需要的debugger要不同，所以原本在做引導實驗11.1時沒有跳出這個error，但到了11.2時需要藍牙新的外接裝置，我們就遇到了這樣的問題，以下是調整Debugger的設定截圖。



1. Sensor Shields

這部分的實驗主要是在使用板子上的 sensor 去感受到外界的溫度、濕度與氣壓。可以看到我們順利將資料匯入板子後，打開 termite 按下 connect，在視窗看到目前的溫度、濕度與氣壓。而視窗更新的資料時間可以藉由我們 hw11 的 code 做變換。



```

1 LAB EXERCISE - Environmental sensors measurements
2
3 In this exercise
4 with the help of
5 Then we will send
6 We can then displ
7
8 At the same time
9
10
11 GOOD LUCK!
12
13
14

```

```

ject: Nucleo_Sensors
'VS.06 update 6 (build 750)',
re compiling one or more files

NUCLEO_F401RE'
nsors.axf" - 0 Error(s), 0 Warn
: 00:00:00
lain\\Desktop\\junior_2\\Embedd
\\build\\Nucleo_Sensors.axf"

```

```

shield (X-NUCLEO-IKS01A1)
the MBED API.
on.
ite).

```

```

does not support C++!

```

```

d at 14:43:25

```

程式架構：

- Initialization
 - Create a DigitalOut objects for the LED
 - Create a Serial objects to communicate via USB
 - Create Ticker objects for the recurring tasks.
 - For blink a LED
 - update the sensor lectures periodically
 - Initialize variables.
- Handlers
 - blinky_handler()
 - Toggle the LED and update the measurements.
 - sensors_handler()
 - Raise a flag that indicates that the measurements need to be read and displayed again.
- Main function

- Attach a function to be called by the Ticker objects at a specific interval in seconds
- In while(1) loop:
 - if the flag in sensors_handler() is high:
 - Read from the sensors (temperature, humidity, pressure)
 - Convert the temperature into Fahrenheit and Kelvin.
 - Send the data to the serial port.
 - Enter sleep mode and wait for interrupts.

心得：

此實驗的sensor偵測環境的溫度、濕度及壓力值，我們有正確的接收到這些環境參數值，也嘗試類似用口對著sensor呼氣去改變這些環境參數，試著看看這個sensor的偵測品質，最後我們發現雖然溫度及濕度有正確的變化，但幅度時大時小，並沒有到非常準確。

2. Your First BLE Program

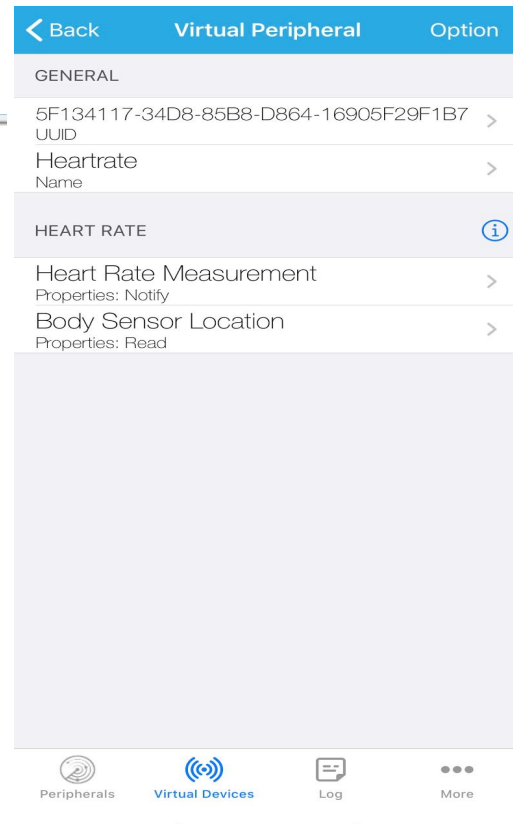
這部分的實驗是將Nucleo-64再加掛載上藍芽傳訊板，來測試keil的藍芽功能，目的是讓我們使用sensor收集到環境的資訊後，能夠在之後的實驗使用藍芽的來將資料傳到手機，不用在透過Termite來讀取sensor所偵測到的資料。



```

Setting up Gatt GATT_INTIMATE_APPL_WHEN_READ_N_WAIT Mask
added bleCharacteristic handle =24
BlueNRGGap::setAdvertisingData
adData[0].length=2
adData[0].AdType=0x1
adData[1].length=7
adData[1].AdType=0x3
Advertising type: COMPLETE_LIST_16BIT_SERVICE_IDS
Advertising type: COMPLETE_LIST_16BIT_SERVICE_IDS
adData[2].length=3
adData[2].AdType=0x19
Advertising type: APPEARANCE
input: deviceAppearance= 0x0 0x40... strlen(deviceAppearance)=2
adData[3].length=12
adData[3].AdType=0x9

```



程式架構：

In main.cpp:

```
class HeartrateDemo: ble:: Gap:: EventHandler
```

- 1.Initialize HeartrateDemo
- 2.Initialize member functions in start()

-
- Callback triggered when the ble initialization process has finished
 - start ble advertising
 - update sensor value
 - toggle LED
 - Event handlers:
 - For disconnection (onDisconnectionComplete)
 - For connection (onConnectionComplete)

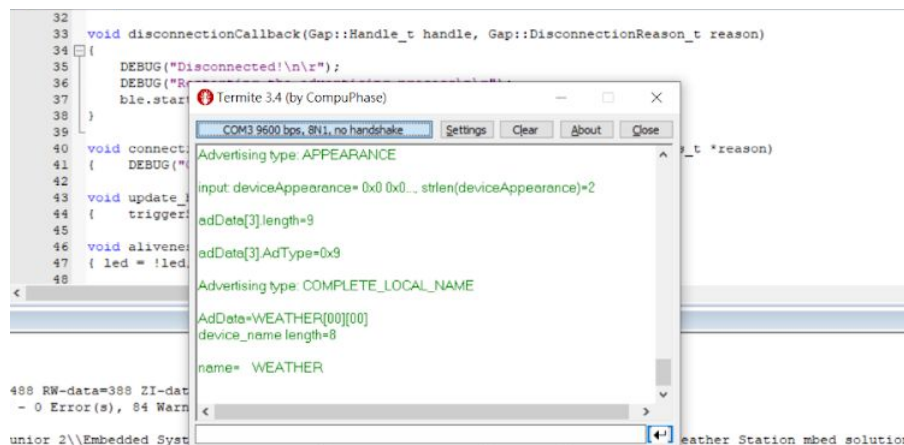
Schedule processing of events from the BLE middleware in the event queue.

心得：

我覺得這個實驗最困難的地方在於，藍芽的傳輸端與手機接收端的連線，一方面在實驗室的環境下，有許多其他裝置的通訊干擾，當時在用手機接收時，我們確定Termite顯示程式是有正常運作的，但是還是沒有找到藍芽名稱的訊號，最後是重新更改code成mbed的版本後，才讓藍芽正確連到手機上，上圖是我們手機正確連上的畫面。

3. Weather Station: part 1

這個部分的實驗為綜合前兩個實驗的成果，11.1的sensor讀取外界資料，11.2的藍芽傳輸，綜合成12的天氣預測站，以下為城市的說明，即在termite, 手機上執行的結果。



程式架構：

- Initialization:
 - BLEDevice
 - Create a DigitalOut objects for the LED
 - Create Ticker objects for the recurring tasks
 - Initialize variables
- Callbacks
 - For disconnection
 - For connection
- Handlers
 - update_handler()

- aliveness_handler()
- Main function:
 - Initialize ble
 - Setup primary services(including humidity, temperature, wind direction, pressure)
 - Setup auxiliary services
 - Setup advertising
 - In while(true) loop:
 - if the flag in update_handler() is high and ble is connected:
 - Read the environmental sensors
 - Calculate the direction where the system is pointing relative to North (so as to know the wind direction)
 - Update the data (humidity, temperature, wind direction, pressure)
 - Enter sleep mode and wait for interrupts.

Weather.java line-49

```
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    mHandler = new Handler();
    mDevicesAdapter = new DevicesAdapter(getLayoutInflater());
```

Weather.java line-82

```
public void onConnectClick(View view) {
    if (mBluetoothAdapter == null) {
        Intent enableBtIntent = new Intent(BluetoothAdapter.ACTION_REQUEST_ENABLE);
        startActivityForResult(enableBtIntent, REQUEST_ENABLE_BT);
    } else {
        String btnText = ((Button) view).getText().toString();
```

心得：

Lab12最困難的一部份在於使藍芽連接成功，一部分在於將Android Studio的app正常運作，我們花了許多時間在研究如何把android studio的code更改參數來使之正確運作，以上是在weather.java中需要調整的兩個參數位置。

而最後我們嘗試在將Android Studio的code導入到手機測試時，因為手機版本的問題，且我們只有一台使用Android系統的手機，導致沒辦法在手機上成功跑出Weather Station的畫面。