

HC-05 藍牙模組 Master Slave 配對

V2020.06.07

目的：

HC-05 藍牙模組配對，Master Slave 之間互相傳送接收資料。

材料清單：

材料	數量
Arduino 開發板	2
USB 訊號線 (A 公對 B 公)	1
HC-05 藍牙模組	2
按鈕開關	2
10K Ω 電阻	2
杜邦線	一些

流程：

1. HC-05 藍牙模組設定 Slave，並找出 ADDR 資料，並紀錄備用
2. HC-05 藍牙模組設定 Master，並指定要連線的 Slave ADDR
3. 簡易硬體規畫
4. 簡易設計 Arduino 程式
5. 測試 Master Slave 配對

1. HC-05 藍牙模組設定 Slave，並顯示 ADDR 資料，紀錄備用

1-1. 藍牙模組，連接 Arduino 開發板

藍牙模組 Key 準備接 Arduino 的 5V，線拉好備用，不接上。

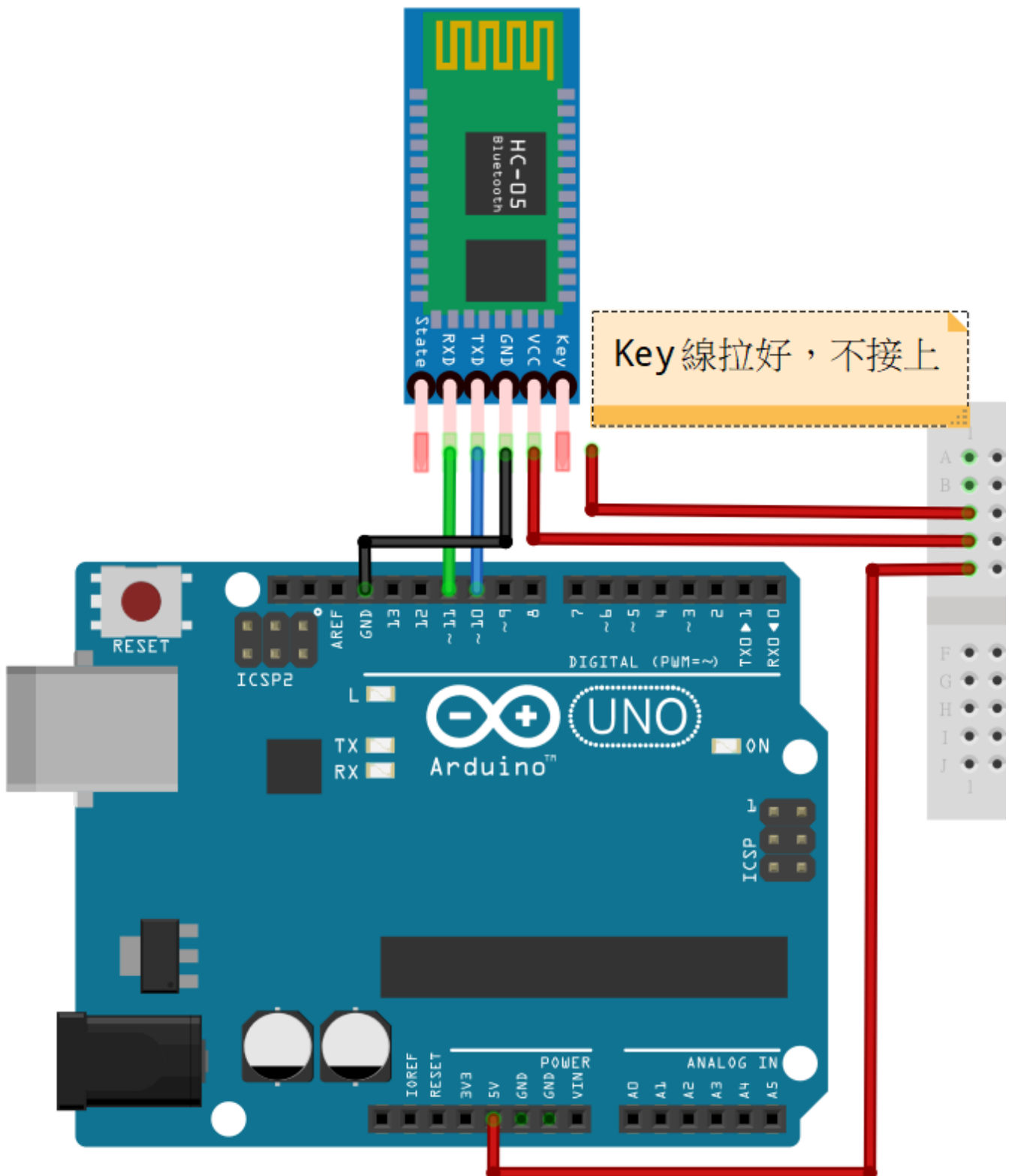
藍牙模組 VCC 接 Arduino 的 5V，(有些用 3.3V，模組背板有標註，或查原廠文件)

藍牙模組 GND 接 Arduino 的 GND

藍牙模組 TX 接 Arduino 的 D10 (軟體 RX)

藍牙模組 RX 接 Arduino 的 D11 (軟體 TX)

藍牙模組 State 不接 (此案例用不到)



電腦 USB 接 Arduino 開發板，上傳 AT mode 程式 BT_AT_HC-05.ino

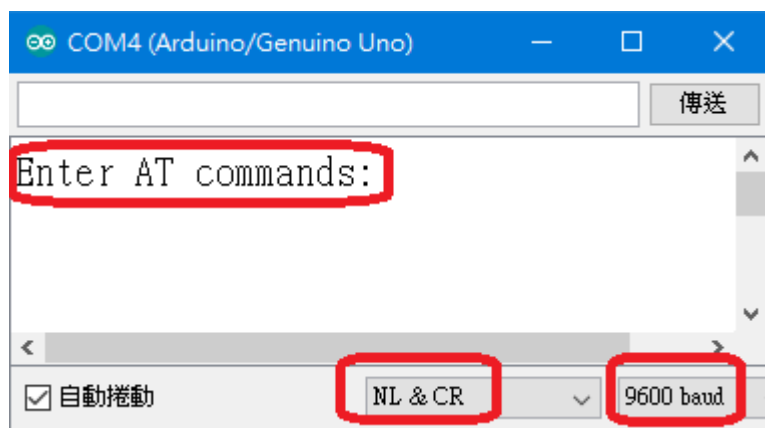
以下 AT mode 程式是由網路下載 (使用 SoftwareSerial)，小弟精簡過：

```
#include <SoftwareSerial.h>
SoftwareSerial BTSerial(10, 11); // RX, TX
void setup()
{
  Serial.begin(9600);
  Serial.println("Enter AT commands:");
  BTSerial.begin(38400); // HC-05 default speed in AT command mode
}
void loop()
{
  // Keep reading from HC-05 and send to Arduino Serial Monitor
  if (BTSerial.available())
    Serial.write(BTSerial.read());

  // Keep reading from Arduino Serial Monitor and send to HC-05
  if (Serial.available())
    BTSerial.write(Serial.read());
}
```

1-2. 進入 AT mode 設定 Slave

打開 Serial Monitor(序列埠監控視窗): 設定 NL & CR，9600 baud



看到 Enter AT commands: 表示程式已準備好進入 AT mode，

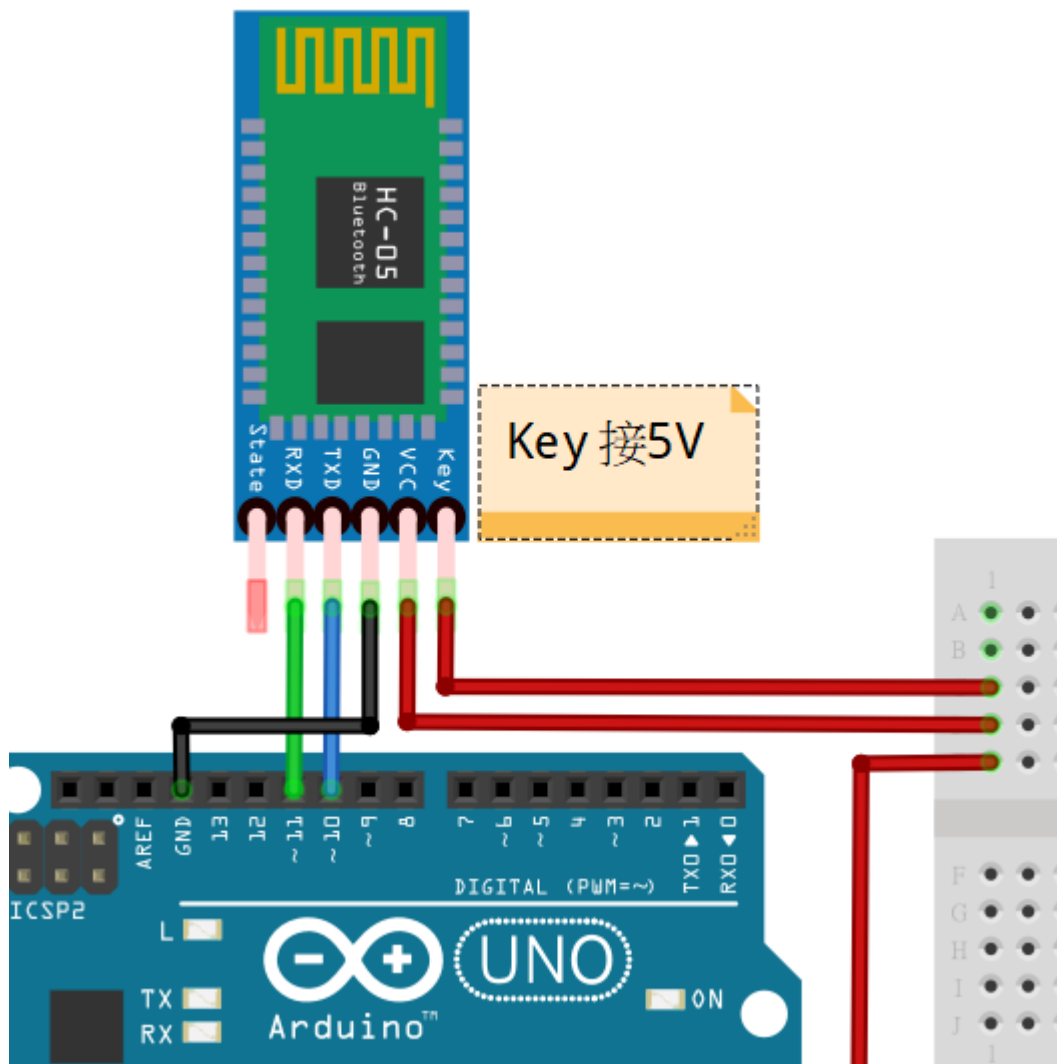
此時將藍牙模組 Key 接上 5V 使藍牙模組進入 AT mode，

如果沒看到 Enter AT commands: 關閉 Serial Monitor 再重開，等待數秒。

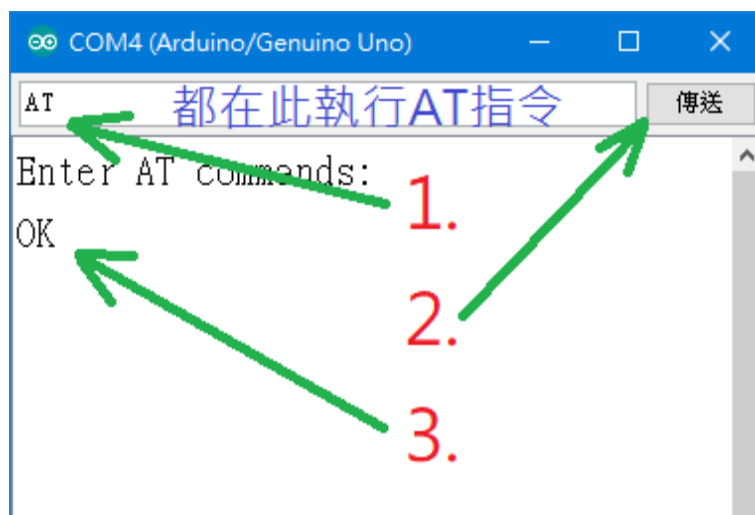
再沒看到，將 Arduino 斷電後拆下 Key pin，重新接上 USB，打開 Serial Monitor，再將 Key pin 接上 5V 使藍牙模組進入 AT mode

註：目前 HC-05 藍牙模組有分 Key pin 和 EN pin 的不同種類

有 EN pin 的 HC-05 藍牙模組，進入 AT mode 的方式是先接 EN pin，先不接 VCC pin，待 Arduino 開發板，程式，Serial Monitor 準備好後，再接上 VCC pin 設定完成後都要取下 Key pin 和 EN pin，之後重新啟動（VCC 重送電）藍牙模組才能使用



輸入大寫 AT 傳送後如果看到回應 OK，就已正確進入 AT mode



執行 AT 指令如果成功只會回應 OK

Slave 設定指令如下：

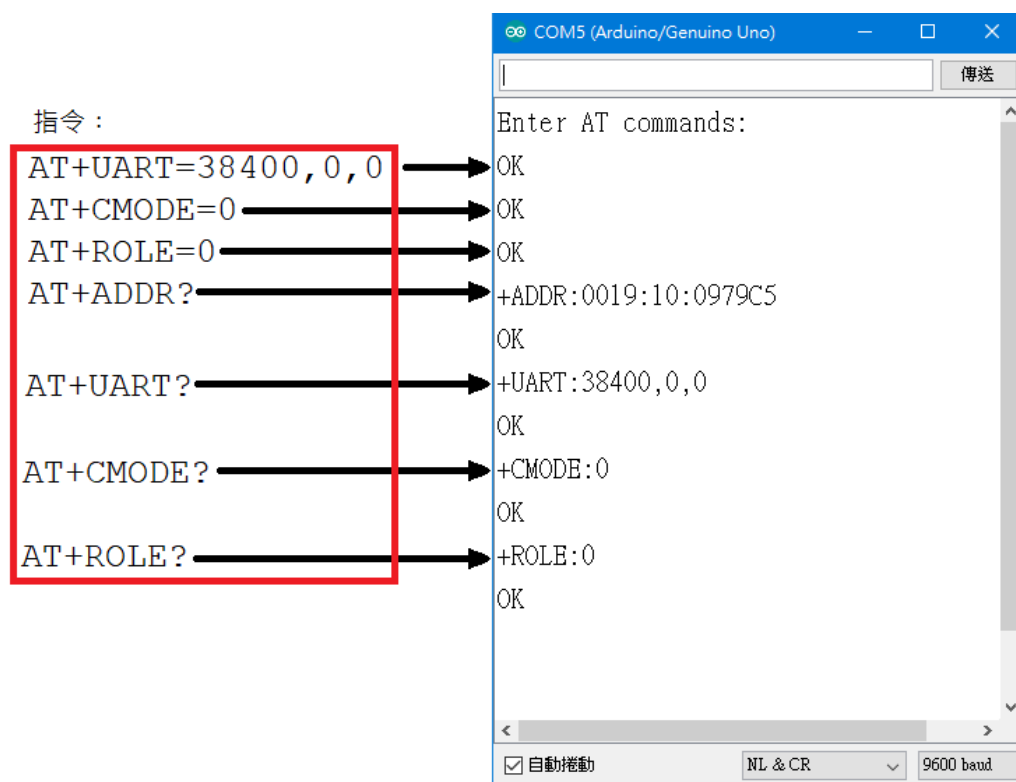
```
AT+UART=38400,0,0
AT+CMODE=0
AT+ROLE=0
```

查詢指令如下，確認設定是否正確

```
AT+UART?
AT+CMODE?
AT+ROLE?
AT+ADDR?
```

AT+ADDR?的回應如： +ADDR:0019:10:0979C5

請紀錄 0019:10:0979C5



設定完成後 **Key pin** 保持空接，就是不接線了，
才能進入一般使用時的 communication mode

2. HC-05 藍牙模組設定 Master，並指定要連線的 Slave ADDR

- 2-1. 取下已設定好的 Slave 藍牙模組，接上另一個藍牙模組設為 Master，接線方式如 1-1
- 2-2. 程式已上傳，不用再重新上傳。
- 2-3. 如 1-3 正確進入 AT mode 後，設定指令 Master 的指令如下：

```
AT+UART=38400,0,0
AT+CMODE=0
AT+ROLE=1
AT+BIND=0019,10,0979C5
```

AT+BIND=0019,10,0979C5 是由之前 Slave 查詢的+ADDR:0019:10:0979C5 修改，

例如：0019:10:0979C5 改為 0019,10,0979C5 (：改為，)

查詢指令如下，確認設定是否正確

```
AT+UART?
AT+CMODE?
AT+ROLE?
AT+BIND?
```

指令：

指令	回應
AT+UART=38400,0,0	OK
AT+CMODE=0	OK
AT+ROLE=1	OK
AT+BIND=0019,10,0979C5	OK
AT+UART?	+UART:38400,0,0 OK
AT+CMODE?	+CMODE:0 OK
AT+ROLE?	+ROLE:1 OK
AT+BIND?	+BIND:0019:10:0979C5 OK

設定完成後 Key pin 保持空接，就是不接線了，
才能進入一般使用時的 communication mode

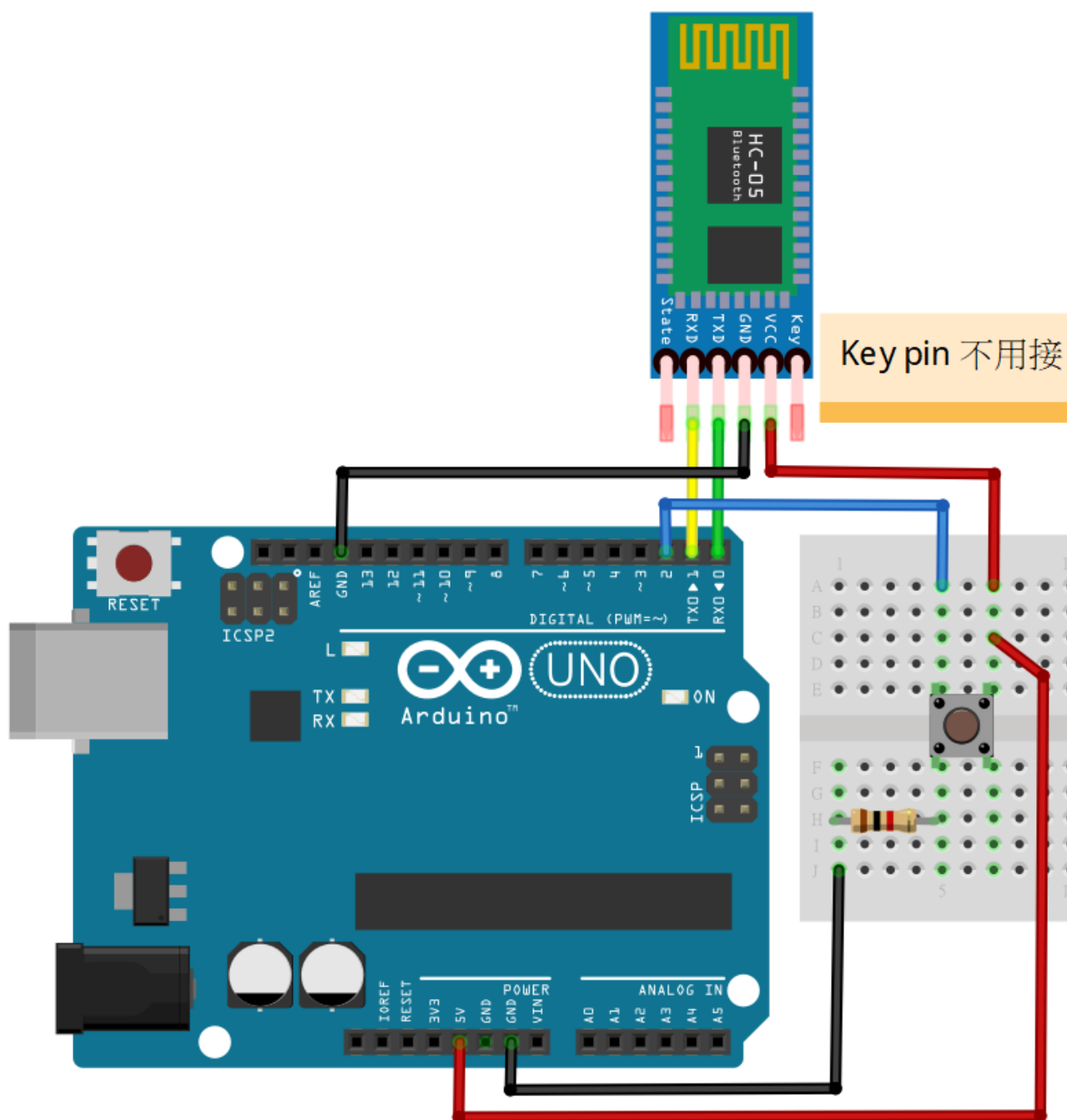
關於 AT command 請看

HC-03/05 Embedded Bluetooth Serial Communication Module AT command set

附檔：HC-05_AT_command.pdf

3. 簡易硬體規畫

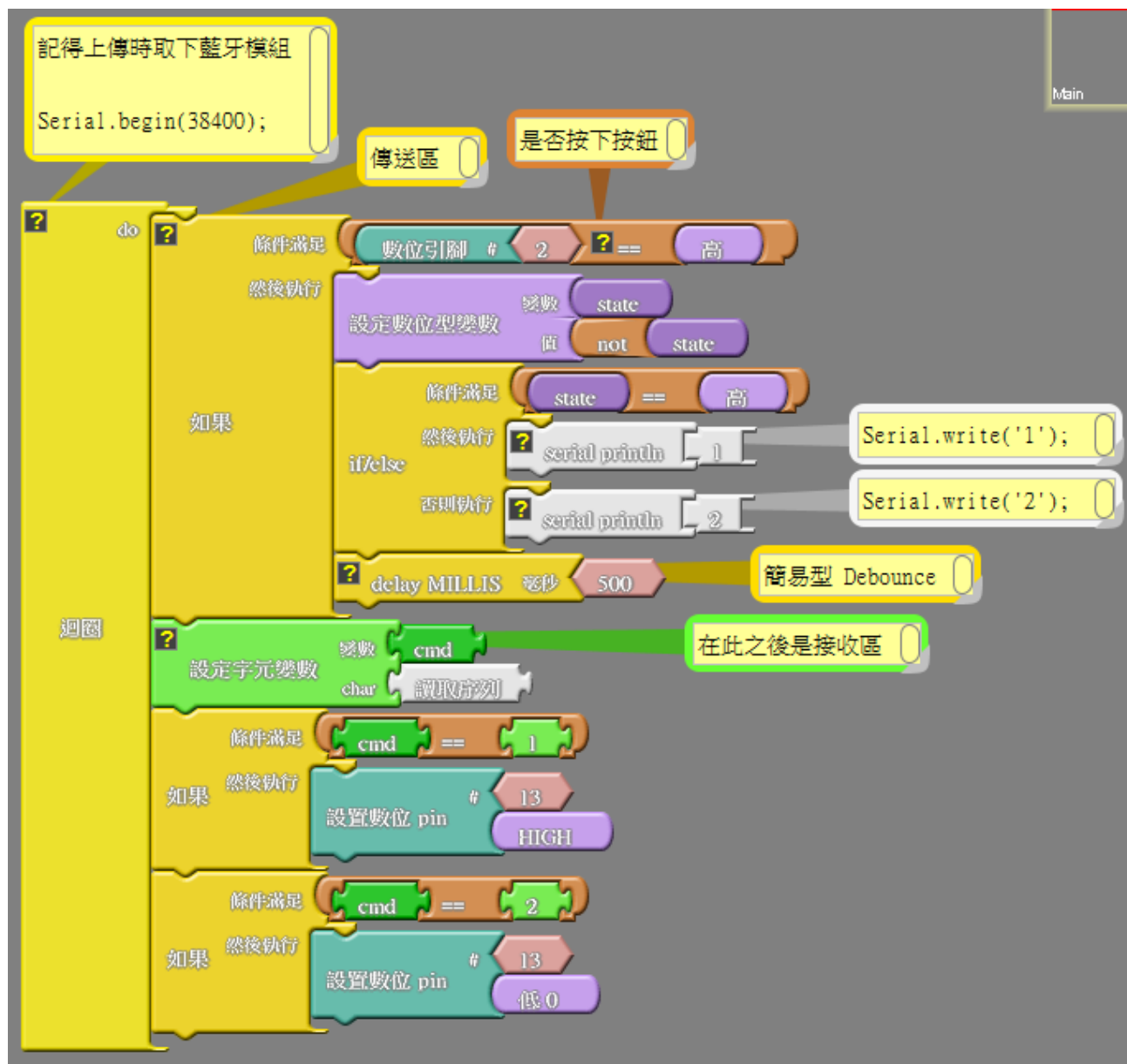
Master, Slave 硬體規畫相同



4. 簡易設計 Arduino 程式

設計目標：Master, Slave 使用同一支程式，用最簡單的方式驗證

ArduBlock 簡易設計如下：（BT_Master_Slave.abp）



先取下藍牙模組 RX, TX 再上傳程式，ArduBlock 上傳後，
在 Arduino 程式碼修改紅色部份如下，並二次上傳（BT_Master_Slave.ino）

```
bool _ABVAR_1_state= false ;
char _ABVAR_2_cmd = ' ' ;

void setup()
{
  pinMode( 2 , INPUT);
  Serial.begin(38400);
  pinMode( 13 , OUTPUT);
}

void loop()
{
  if (( ( digitalRead(2) ) == ( HIGH ) ))
  {
    _ABVAR_1_state = !( _ABVAR_1_state ) ;
    if (( ( _ABVAR_1_state ) == ( HIGH ) ))
    {
      Serial.write('1');
    }
    else
    {
      Serial.write('1');
    }
    delay( 500 );
  }
  _ABVAR_2_cmd = Serial.read();
  if (( ( _ABVAR_2_cmd ) == ( '1' ) ))
  {
    digitalWrite( 13 , HIGH );
  }
  if (( ( _ABVAR_2_cmd ) == ( '2' ) ))
  {
    digitalWrite( 13 , LOW );
  }
}
```

上傳後就可以將 藍牙模組 RX, TX 接回

因為硬體規畫都相同，所以第二個 Arduino 開發板也用同一程式，
記得先取下藍牙模組 RX, TX 再上傳程式

5. 測試 Master Slave 配對

如果有 I/O 擴展板，依其方式連接，先取下藍牙模組 RX, TX 再上傳程式

以上 AT command 設定，硬體接線，上傳軟體 如果沒有問題，

兩組 Arduino 供電後，藍牙模組 LED 一直快閃，

Master, Slave 自動連線後，2~3 秒快閃 2 下循環，表示連線成功

待 Arduino 完成開機，正常執行程式後，

按下 Master 按鈕開關，Slave Arduino 板上的 D13 LED 亮，再按一下按鈕開關 LED 滅，

按 Slave 按鈕開關 Master Arduino 板上的 D13 LED 亮，再按一下按鈕開關 LED 滅

以上演示了 Arduino 藍牙 Master, Slave 無線通訊互傳資料簡單的方式，（雙向）

會傳一個資訊，多傳幾個應該沒什麼問題，

再套上其他感測器，多層次的程式，就可以有多功能

參考資料：

玢宇老師的生活科技網站

<https://livingtech.education/2018/06/03/arduino%e5%af%a6%e7%94%a8-hc-05-at-mode%e4%b8%bb%e5%be%9e%e9%85%8d%e5%b0%8d/>

雄

<https://gsyan888.blogspot.com/2015/03/arduino-hc-05-master-and-slave.html>

超圖解系列圖書

<https://swf.com.tw/?p=693>

<https://swf.com.tw/?p=705>

<https://swf.com.tw/?p=712>

<https://swf.com.tw/?p=750>

<https://swf.com.tw/?p=759>

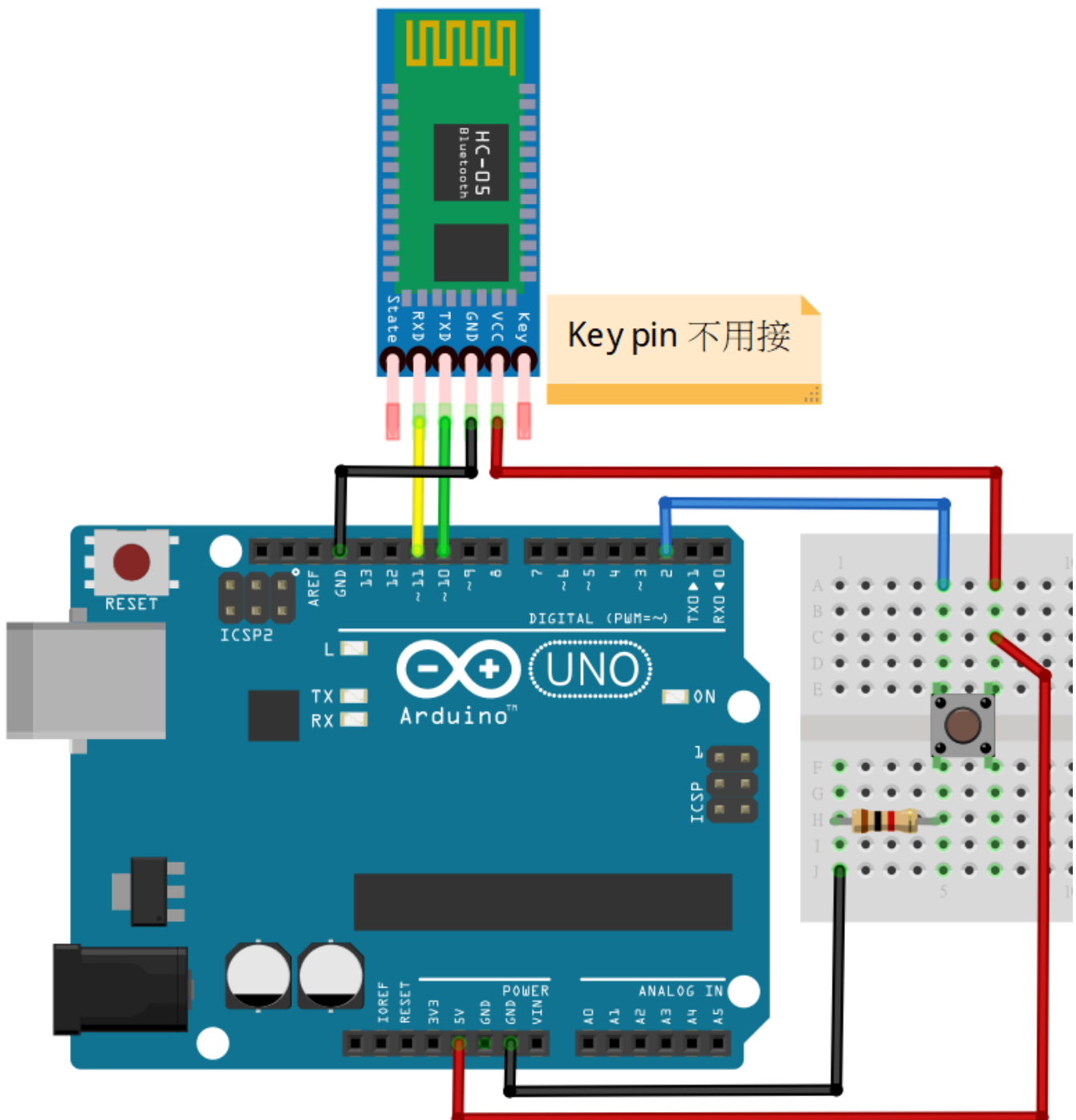
後記：

開發程式時要經常上傳程式測試，一直要拆模組，接模組，拆模組，接模組.....實在麻煩，所以用 SoftwareSerial 函式庫，可以不用拆裝模組，但是要佔用 2 個 digital pin，在此用了 D10，D11 做軟體 RX, TX

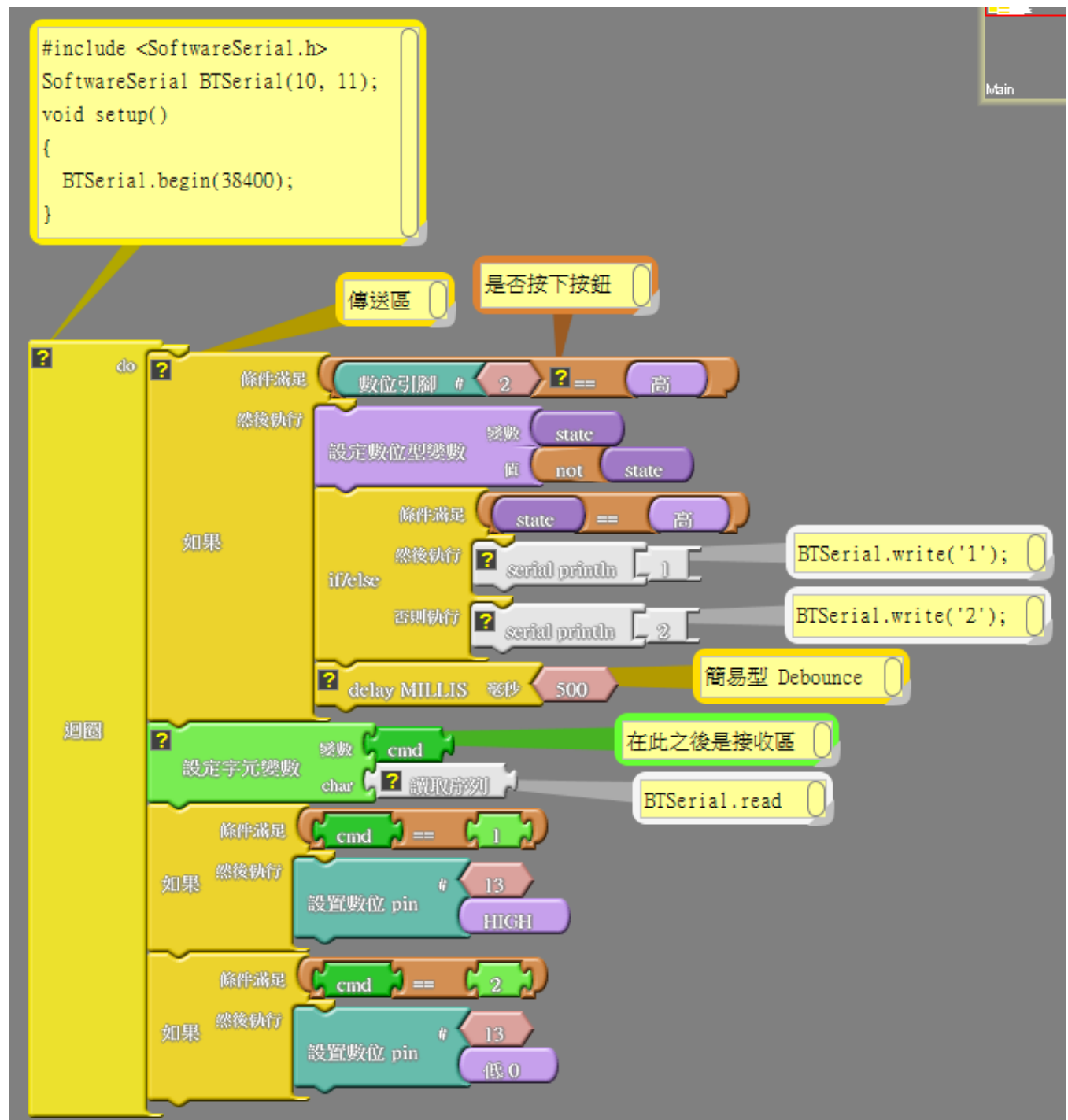
Serial Monitor，上傳程式與 Arduino 通訊用了 RX (D0)，TX(D1)

以 SoftwareSerial 函式庫，定義 Arduino D10 為軟體 RX，D11 為軟體 TX，給藍牙模組與 Arduino 通訊

硬體接線：



ArduBlock 程式如下：其實和之前程式相同 (BT_Master_Slave_SoftwareSerial.abp)



ArduBlock 上傳後，在 Arduino 程式碼修改部份和上面不同，

新增及修改紅色部份如下，並二次上傳 （BT_Master_Slave_SoftwareSerial.ino）

```
#include <SoftwareSerial.h>
SoftwareSerial BTSerial(10, 11);
bool _ABVAR_1_state= false ;
char _ABVAR_2_cmd = ' ' ;

void setup()
{
  pinMode( 2 , INPUT);
  Serial.begin(9600);
  pinMode( 13 , OUTPUT);
  BTSerial.begin(38400);
}

void loop()
{
  if (( ( digitalRead(2) ) == ( HIGH ) ))
  {
    _ABVAR_1_state = !( _ABVAR_1_state ) ;
    if (( ( _ABVAR_1_state ) == ( HIGH ) ))
    {
      BTSerial.write('1');
    }
    else
    {
      BTSerial.write('2');
    }
  }
  delay( 500 );
  _ABVAR_2_cmd = BTSerial.read();
  if (( ( _ABVAR_2_cmd ) == ( '1' ) ))
  {
    digitalWrite( 13 , HIGH );
  }
  if (( ( _ABVAR_2_cmd ) == ( '2' ) ))
  {
    digitalWrite( 13 , LOW );
  }
}
```

因為硬體規畫都相同，所以第二個 Arduino 開發板也用同一程式，直接上傳就好。

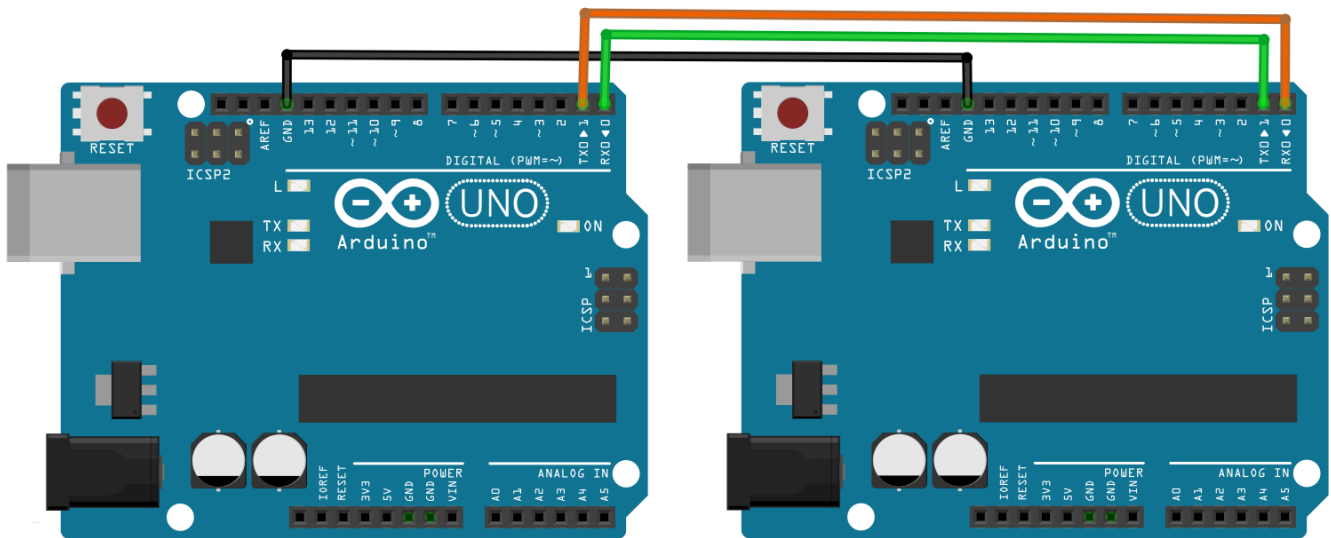
不同的是不用再拆接藍牙模組 RX, TX

- 註 1：可修改程式定義 RX/TX 腳位，並接好藍牙模組對應 Arduino 腳位（我試過 D7,D8 正常）
- 註 2：有些藍牙模組 VCC 使用 3.3V，經測 5V 在設定階段正常，長久使用時建議改接 3.3V，有些藍牙模組 VCC 使用 5V

再後記：

假如我暫時沒有藍牙模組可用，也想體驗一下，是可行的！

兩個Arduino對連，RX接對方TX，TX接對方RX，GND接對方GND(共地)，可以在沒藍牙模組時，用有線方式開發程式，到時設定好藍牙模組，接上立即可用 Serial.read Serial.write 互傳資訊，上傳程式時，要將此線移除，上傳完成再接上



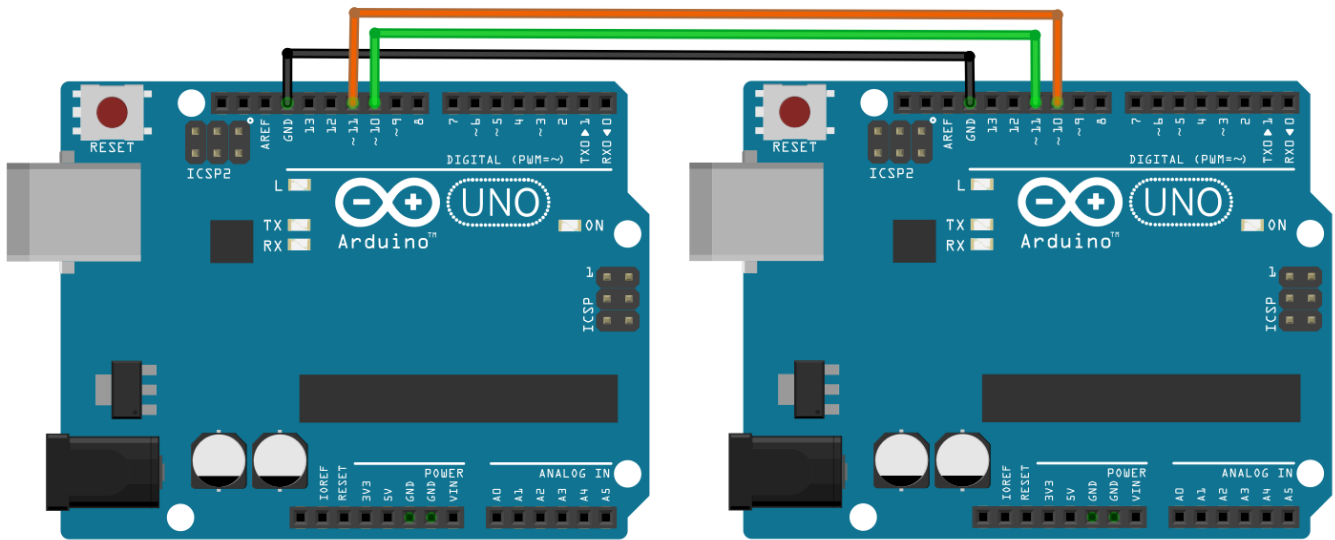
上圖只有畫出取代 HC-05 藍牙模組部份，按鈕開關感測器和之前一樣不變
一樣上傳程式時要取下 RX, TX

兩組 Arduino 程式和之前一樣 (BT_Master_Slave.ino)

再再後記：

假如我暫時沒有藍牙模組可用，想體驗一下，不想上傳程式時一直拆接線，也是可行的！

使用SoftwareSerial定義 D10為RX , D11為TX，兩個Arduino對連，RX接對方TX，TX接對方RX，GND接對方GND(共地)，可以在沒藍牙模組時，用有線方式開發程式，到時設定好藍牙模組，接上立即可用，可以隨時上傳程式，不用拆接線



也是只畫出取代 HC-05 藍牙模組部份，按鈕開關感測器和之前一樣不變
程式和“後記”使用 SoftwareSerial 一樣 (BT_Master_Slave_SoftwareSerial.ino)