

目 錄

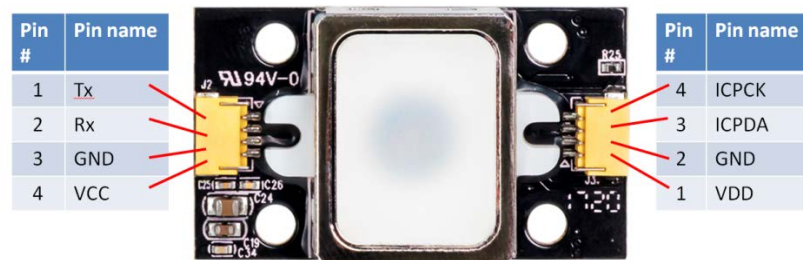
腳位功能	2
降低待機功耗的做法.....	3
與 Arduino Uno 連接使用.....	3
邏輯位準轉換電路.....	3
接線	4
下載並執行範例程式	5
指紋登錄.....	5
指紋辨識.....	6
查詢已登錄的指紋數目	7
刪除 ID	7
使用 SDK_DEMO 軟體	9
接線	9
下載並執行 SDK_DEMO.....	10
查看模組詳細資料	11
指紋登錄.....	11
指紋驗證.....	12
指紋辨識.....	13

腳位功能

指紋辨識模組 GTM-5210F52 上有 2 個 4-pin 的 JST-SH 連接器(腳距 1.0mm) · 分別為 UART 連接器(UART Connector)及觸控 IC 連接器(Touch IC Connector) · 腳位如下圖所示。

UART Connector

Touch IC Connector



UART 連接器的腳位說明：

Pin No.	Pin Name	說明
1	Tx	模組 UART 介面的資料傳送腳，電壓位準為 3.3V(TTL)
2	Rx	模組 UART 介面的資料接收腳，電壓位準為 3.3V(TTL)
3	GND	接地
4	VCC	電源，接 3.3V ~ 6V DC，此電源僅用來供電給指紋辨識模組，模組內建有穩壓器，此電源經過穩壓器再供電給整個模組(但不包括觸控 IC)

觸控 IC 連接器的腳位說明：

Pin No.	Pin Name	說明
1	VDD	電源，接 3.3V DC，此電源僅用來供電給觸控 IC，不供電給模組的其他電路
2	GND	接地
3	ICPDA	平常模式下，未使用。 燒錄模式下，為觸控 IC 的燒錄腳位。
4	ICPCK	平常模式下，做為觸控 IC 的輸出腳，當手指碰觸指紋感測器的周圍的鐵框時，輸出高電平。 燒錄模式下，為觸控 IC 的燒錄腳位。

註：本產品出貨時觸控 IC 已燒錄程式，使用者無需燒錄觸控 IC。

本產品出貨時未附有線材，請您自行購買，將訊號引出使用。

降低待機功耗的做法

模組內建觸控 IC，其主要功能是偵測手指是否接觸到指紋感測器，當手指碰觸指紋感測器的周圍的鐵框時，觸控 IC 連接器的 pin 4 (ICPCK) 會輸出高電平，若無接觸則為低電位。

在日常的操作裡，我們可以在偵測到有手指按壓在指紋感測器上時再供電給指紋辨識模組，如此可以降低模組待機時的功耗。我們可以使用外接的電路或微控制器(例如 Arduino 板)在偵測到 ICPCK 腳位為高電平時再供電給整個模組(透過 UART 連接器 pin 4 VCC 腳)。

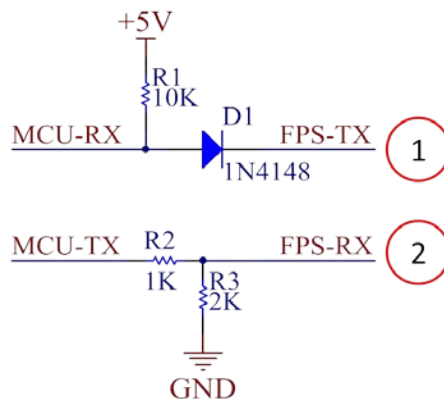
若要採用此作法來節省耗電，須注意必須保持供電給觸控 IC 以使其正常工作。如果您不需要此功能，可以不使用觸控 IC 接口，此接口不接線並不影響整個指紋辨識模組的正常工作，但此時須保持 UART 接口 pin 4 VCC 腳持續供電(3.3V ~ 6V)的狀態。

與 Arduino Uno 連接使用

此模組有豐富的指令集，外部控制器可使用這些指令來操作本模組。以下介紹使用 Arduino Uno 來操作指紋辨識模組 GTM-5210F52。

邏輯位準轉換電路

由於 Arduino Uno 板為 5V 工作電壓而指紋辨識模組的 UART 電壓位準為 3.3V，因此我們需要做 5V 與 3.3V 的邏輯位準轉換，下圖是一個簡單的 5V 與 3.3V TTL 位準轉換電路。



5V/3.3V logic Level Converter Circuit

接線

配合範例程式使用時，請依以下的接線方式。

Arduino UNO 板的接線：

Pin	功能	接線方式
4	Software serial RX	接到邏輯位準轉換電路的 MCU-RX
5	Software serial TX	接到邏輯位準轉換電路的 MCU-TX
6	手指按壓偵測	接到指紋辨識模組 Touch IC 接口的 ICPCCK (pin 4) · 當手指觸摸到模組感測器的鐵框時，ICPCCK 腳位為 High，未接觸時為 Low [註:我們提供的範例程式未使用到此訊號，可以不接此線]
5V	提供 5V 電源	接到指紋辨識模組 UART 連接器的 VCC (pin 4) · 提供電源給指紋辨識模組，同時也接到邏輯位準轉換電路的 +5V
3.3V	提供 3.3V 電源	接到指紋辨識模組 Touch IC 接口的 VDD (pin 1) · 提供電源給指紋辨識模組裡的 Touch IC
GND	接地	接到指紋辨識模組 UART 接口的 GND (pin 3) · Touch IC 接口的 GND (pin 2) · 同時也接到邏輯位準轉換電路的 GND

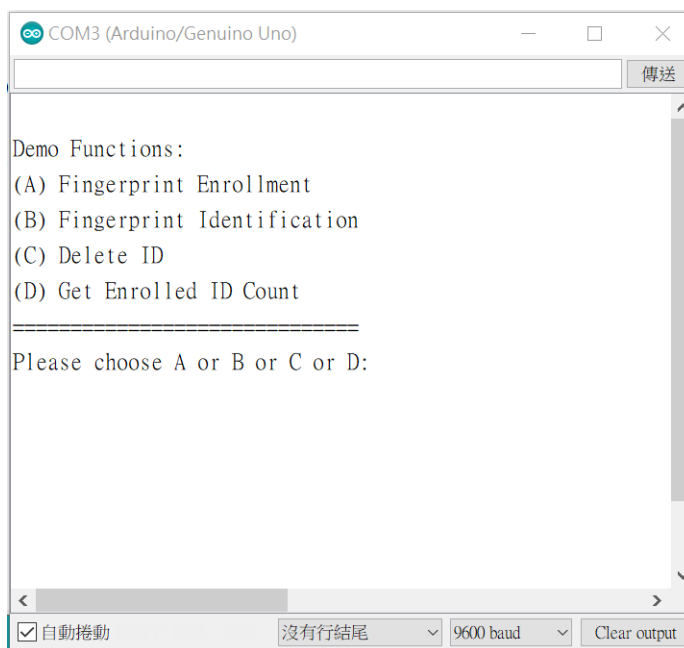
指紋辨識模組的接線：

連接器	Pin	Pin Name	接線方式
UART 連接器	1	TX	接到邏輯位準轉換電路的 FTP-TX
	2	RX	接到邏輯位準轉換電路的 FTP-RX
	3	GND	接到 Arduino UNO 板的 GND
	4	VCC	接到 Arduino UNO 板的 5V
Touch IC 連接器	1	VDD	VDD 接到 Arduino UNO 板的 3.3V
	2	GND	接到 Arduino UNO 板的 GND
	3	ICPDA	無
	4	ICPCCK	接到 Arduino UNO 板的 pin 6 [註:我們提供的範例程式未使用到此訊號，可以不接此線]

下載並執行範例程式

網路上的開源社群已經有針對此系列指紋辨識模組的 Arduino 開源程式[\[鏈結\]](#)，您可以參考，我們基於此開源程式開發了範例程式 `FPS_GTM5210F52_demo`，請在本產品產品頁面的 DOCUMENTS 下載，下載後將其加入開源程式的 examples 目錄內。

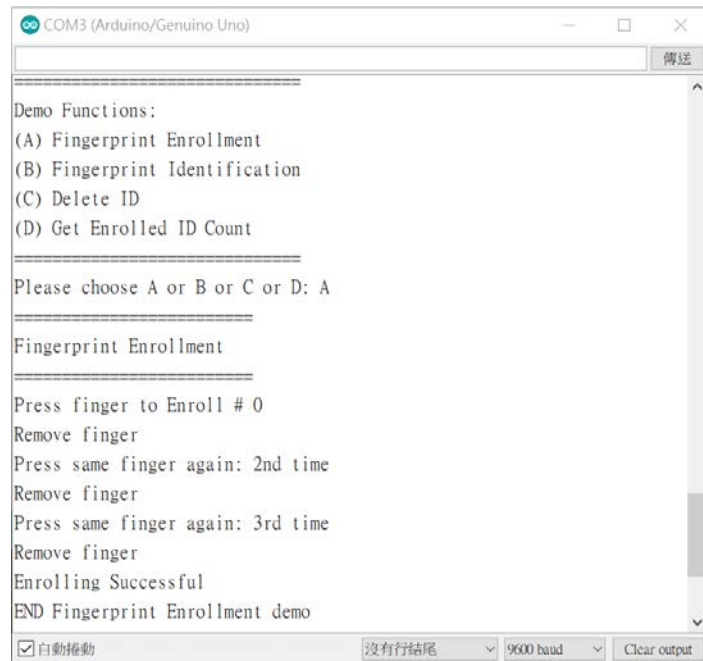
進入 Arduino IDE 後，打開工具 → 序列埠監控視窗 (Ctrl+Shift+M)，顯示初始畫面如下，範例程式演示四個功能：指紋登錄、指紋辨識、刪除 ID、查詢已登錄的指紋數目。



指紋登錄

輸入 A 可執行指紋登錄，範例程式會從 0 開始找到一個尚未被使用的 ID 來做為此次登錄的 ID，每次登錄一個指紋須按壓 3 次，依照畫面的提示訊息進行按壓手指及移開手指即可。

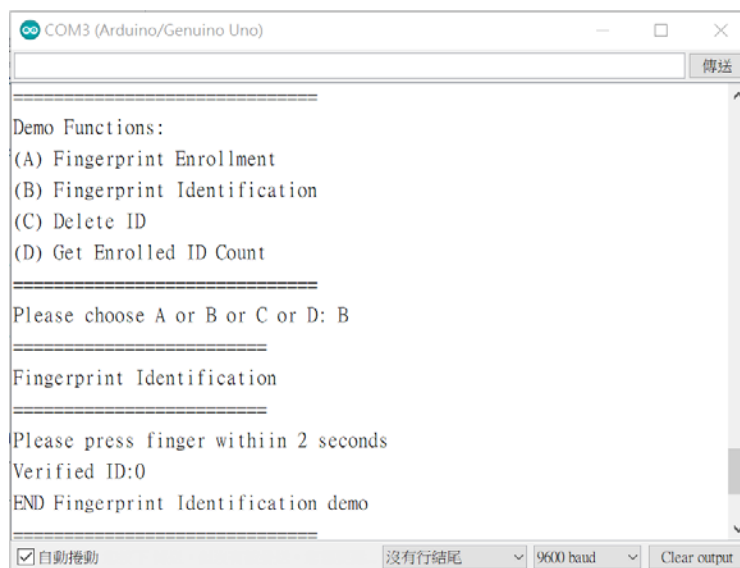
指紋登錄程式主要使用指紋識別模組的 `CaptureFinger` 指令及 `Enrollx` 指令 (`Enroll1`、`Enroll2`、`Enroll3`)。 `CaptureFinger` 指令的功能為取指紋，如果在執行 `CaptureFinger` 指令時手指未按壓，則顯示錯誤訊息 `Fail to capture x finger` (x 為 `first`、`second`、`third`)。 `Enrollx` 指令的功能為產生指紋樣板檔 (`template`)，如果指令執行失敗則顯示錯誤訊息 `Enrolling Failed with error code: x`； $x=1$ 及 $x=2$ 都是代表產出樣板檔失敗，可能的原因是按壓力道太輕，指紋不夠清晰。 $x=3$ 代表指紋已經登錄過了，在資料庫中存在相同指紋，您可使用 `Delete ID` 功能來移除指紋。



```
COM3 (Arduino/Genuino Uno)
=====
Demo Functions:
(A) Fingerprint Enrollment
(B) Fingerprint Identification
(C) Delete ID
(D) Get Enrolled ID Count
=====
Please choose A or B or C or D: A
=====
Fingerprint Enrollment
=====
Press finger to Enroll # 0
Remove finger
Press same finger again: 2nd time
Remove finger
Press same finger again: 3rd time
Remove finger
Enrolling Successful
END Fingerprint Enrollment demo
=====
 自動捲動  沒有行結尾  9600 baud  Clear output
```

指紋辨識

輸入 B 可執行指紋辨識，若辨識成功會顯示該指紋的 ID 號碼。

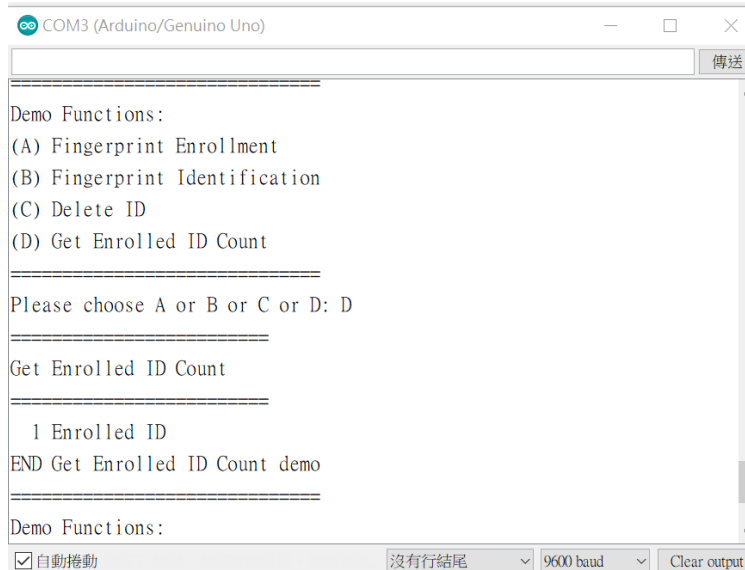


```
COM3 (Arduino/Genuino Uno)
=====
Demo Functions:
(A) Fingerprint Enrollment
(B) Fingerprint Identification
(C) Delete ID
(D) Get Enrolled ID Count
=====
Please choose A or B or C or D: B
=====
Fingerprint Identification
=====
Please press finger within 2 seconds
Verified ID:0
END Fingerprint Identification demo
=====
 自動捲動  沒有行結尾  9600 baud  Clear output
```

範例程式的設計是點亮 LED 後等待 2 秒再取指紋，您可自行修改程式調整等待時間。指紋辨識程式主要使用指紋識別模組的 CaptureFinger 指令及 Identify 指令，若在取指紋時失敗，則顯示錯誤訊息 Please press finger，未按壓手指。若辨識失敗則顯示錯誤訊息 Finger not Found，原因是此指紋未登錄過。

查詢已登錄的指紋數目

輸入 **D** 可查詢已登錄的指紋數目。

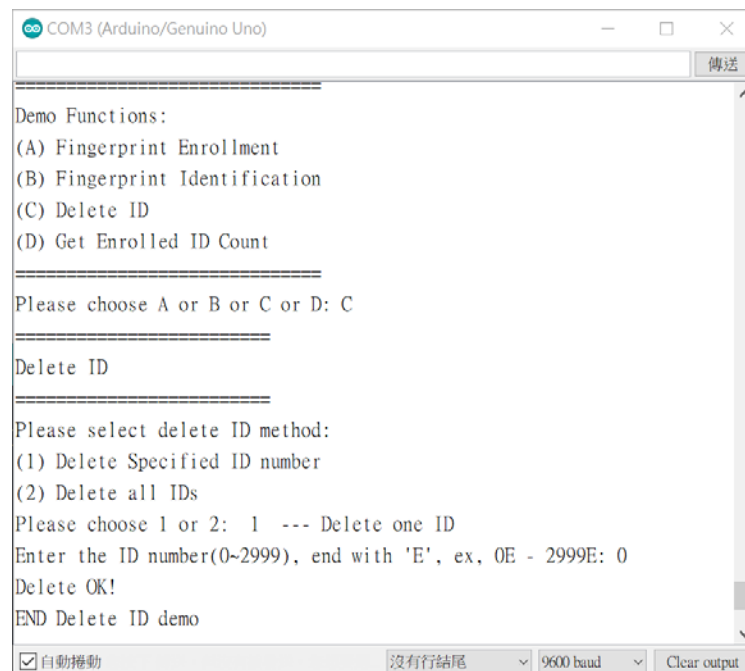


```
COM3 (Arduino/Genuino Uno)
Demo Functions:
(A) Fingerprint Enrollment
(B) Fingerprint Identification
(C) Delete ID
(D) Get Enrolled ID Count
=====
Please choose A or B or C or D: D
=====
Get Enrolled ID Count
=====
1 Enrolled ID
END Get Enrolled ID Count demo
=====
Demo Functions:
```

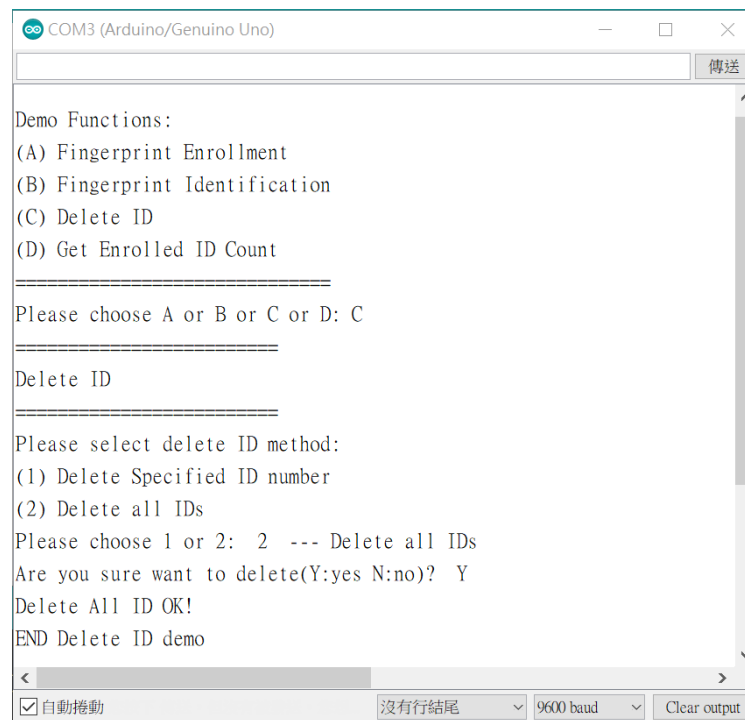
此功能主要使用指紋識別模組的 `GetEnrollCount` 指令，用以取得模組上的資料庫目前儲存的指紋數目。

刪除 ID

輸入 **C** 可執行刪除 ID，可選擇刪除指定的 ID 或是刪除所有 ID 回復到空白狀態。依範例程式的設計，指定 ID 時，請注意在 ID 號碼後加上 **E** 才能成功。



```
COM3 (Arduino/Genuino Uno)
Demo Functions:
(A) Fingerprint Enrollment
(B) Fingerprint Identification
(C) Delete ID
(D) Get Enrolled ID Count
=====
Please choose A or B or C or D: C
=====
Delete ID
=====
Please select delete ID method:
(1) Delete Specified ID number
(2) Delete all IDs
Please choose 1 or 2: 1 --- Delete one ID
Enter the ID number(0~2999), end with 'E', ex, 0E - 2999E: 0
Delete OK!
END Delete ID demo
```



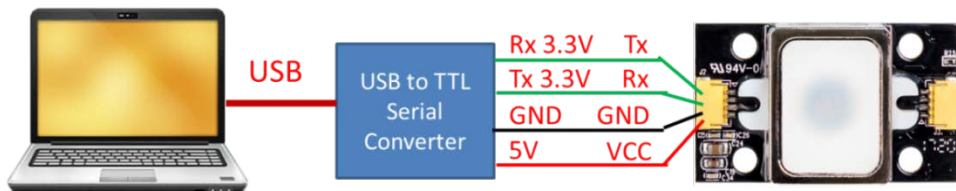
此功能主要使用指紋識別模組的 DeleteID 及 DeleteAll 指令。

使用 SDK_DEMO 軟體

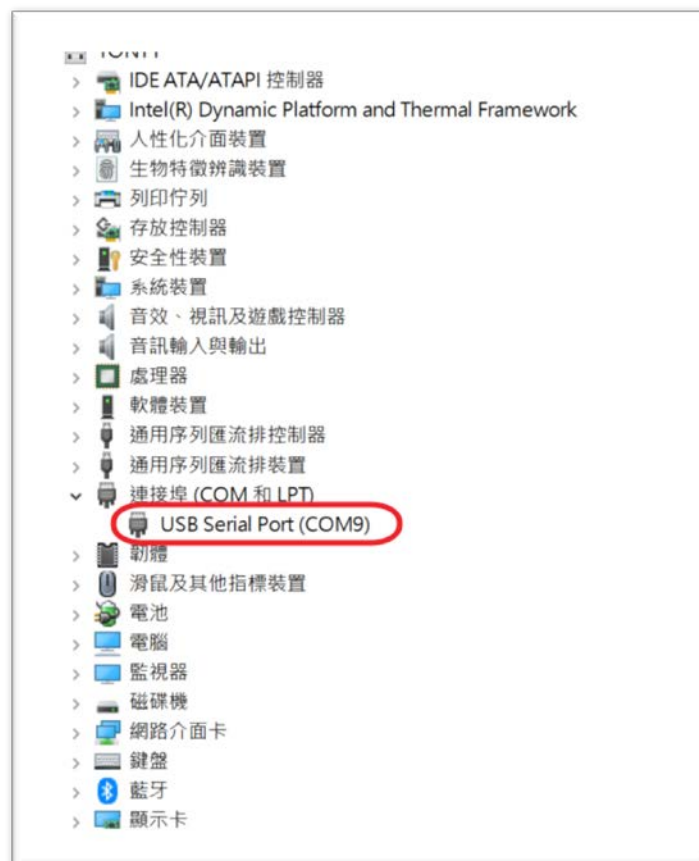
SDK_DEMO 是一個在電腦上執行的軟體，它可以演示指紋辨識模組 GTM-5210F52 的大部分基本功能。您只需要正確地將模組連線，不需要寫程式即可開始操作指紋辨識模組，SDK_DEMO 程式可以在產品頁面的 DOCUMENTS 裡下載。

接線

我們使用市售的 USB 轉 TTL 模組來連接指紋辨識模組，由於指紋辨識模組的 UART 是 3.3V 位準，所以設定 USB 轉 TTL 模組工作在 3.3V，但把 5V 電源單獨拉出供電給指紋辨識模組(接到 UART 接口的 pin 4 VCC 腳)。如果 USB 轉 TTL 模組的 5V 輸出電流不足，則必須將 VCC 腳單獨供電，電壓範圍 3.3V~6V。

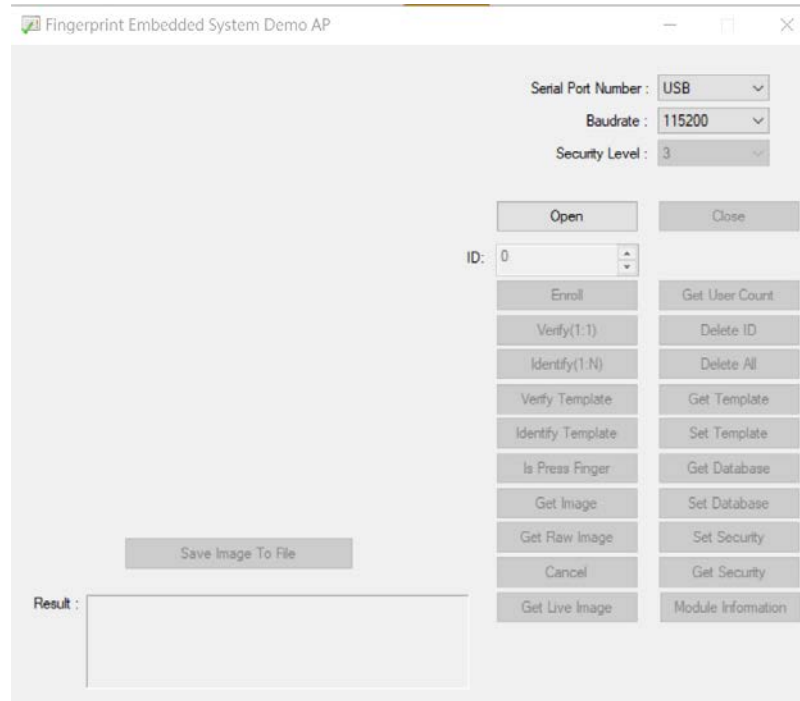


完成接線之後，在 windows 的裝置管理員裡可查看 USB 轉 TTL 模組被指定的 COM 埠，請注意這個 COM 埠不是固定的，它會隨著電腦的配置而改變。

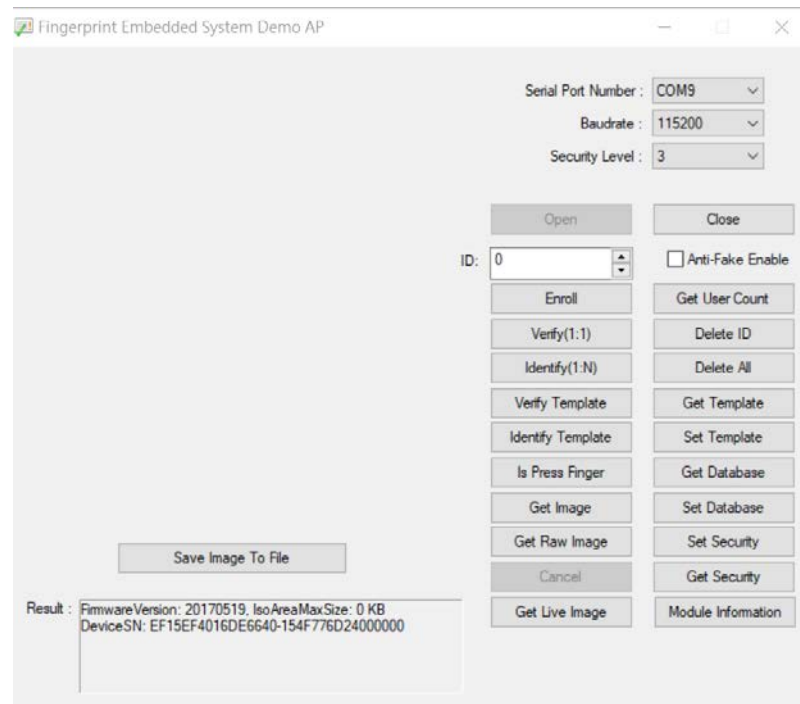


下載並執行 SDK_DEMO

啟動 SDK_DEMO 後的畫面如下:

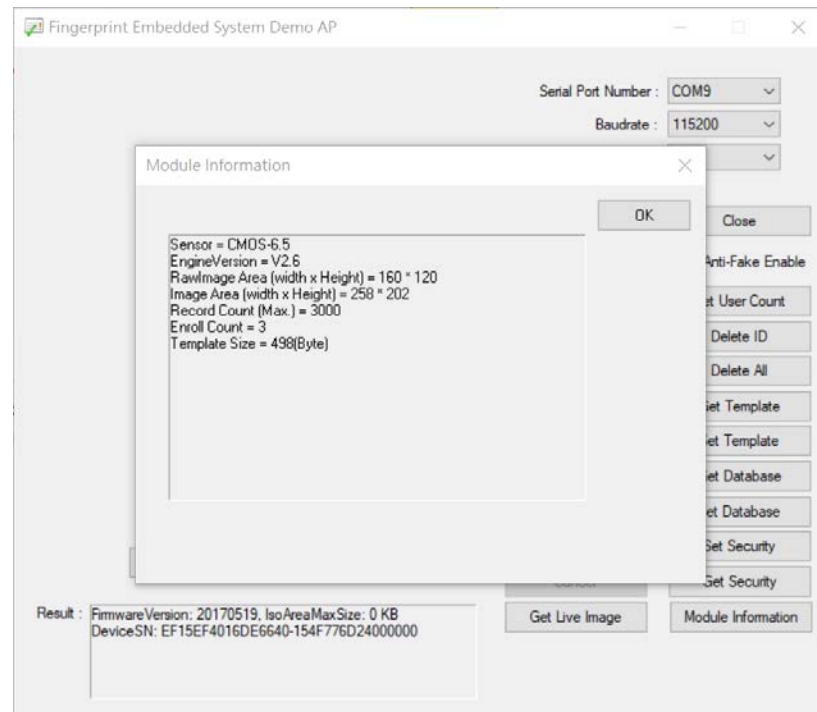


修改右上方 Serial Port Number 後再點擊 Open 即可啟動模組。



查看模組詳細資料

您可以點擊右下角的 **Module Information** 查看模組的詳細資料。

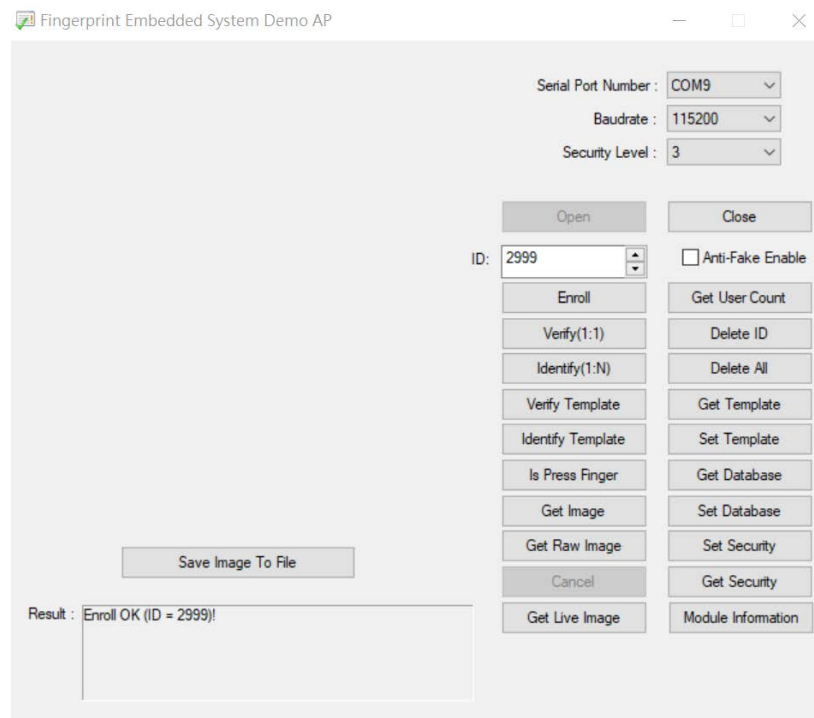


指紋登錄

點擊 **Enroll** 以登錄指紋，點擊前，可以更改 **ID** 欄位裡的數值以指定此次登錄的指紋的 ID 編號。本模組的容量為 3000 個指紋，ID 編號由 0 到 2999，此次登錄我們設定 ID 編號為 2999。

點擊 **Enroll** 後，指紋辨識模組會亮起。‘**Result:**’ 欄位裡顯示“input finger 1 !”的訊息，請您按壓手指於感測器的平板上，成功登錄一個指紋需要採樣三次，依照‘**Result:**’ 欄位的指示完成登錄。

本模組支持 360 度辨識，按壓手指時不限於從哪個方向按壓，都可以完成採樣及辨識。



指紋驗證

Verify(1:1)的功能是比對在 ID 欄位所指定的 ID 編號與按壓的手指是否相同？點擊 Verify(1:1)後，程式等待您按壓手指，若您按壓正確的手指，畫面會顯示 ID 及此次驗證所耗用的時間。



指紋辨識

Identify(1:N)的功能是比對您所按壓的手指與已登錄的指紋資料庫，找出其 ID 編號，若您按壓的手指並未登錄過，畫面會顯示 NG.

