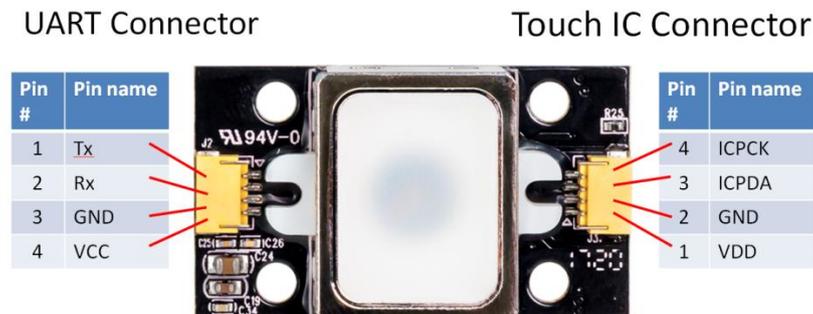


目 录

脚位功能	2
降低待机功耗的做法	3
与 Arduino Uno 连接使用	3
接线	4
下载并执行范例程序	5
指纹登记	5
指纹识别	6
查询已登录的指纹数目	7
删除 ID	7
使用 SDK_DEMO 软件	9
接线	9
下载并执行 SDK_DEMO	10
查看模块详细数据	11
指纹登记	11
指纹验证	12
指纹识别	13

脚位功能

指纹识别模块 GTM-5210F332 上有 2 个 4-pin 的 JST-SH 连接器(脚距 1.0mm)，分别为 UART 连接器(UART Connector)及触控 IC 连接器(Touch IC Connector)，引脚如下图所示。



UART 连接器的引脚说明：

Pin No.	Pin Name	说明
1	Tx	模块 UART 接口的数据传送脚，电压为 3.3V(TTL)
2	Rx	模块 UART 接口的数据接收脚，电压为 3.3V(TTL)
3	GND	接地
4	VCC	电源, 接 3.3V ~ 6V DC, 此电源仅用来供电给指纹识别模块, 模块内建有稳压器, 此电源经过稳压器再供电给整个模块(但不包括触控 IC)

触控 IC 连接器的引脚说明：

Pin No.	Pin Name	说明
1	VDD	电源, 接 3.3V DC, 此电源仅用来供电给触控 IC, 不供电给模块的其他电路
2	GND	接地
3	ICPDA	平常模式下, 未使用。 烧录模式下, 为触控 IC 的烧录引脚。
4	ICPCK	平常模式下, 作为触控 IC 的输出脚, 当手指碰触指纹传感器的周围的铁框时, 输出高电平。 烧录模式下, 为触控 IC 的烧录引脚。

注：本产品出货时触控 IC 已烧录程序，用户无需烧录触控 IC。

本产品出货时未附有线材，请您自行购买，将讯号引出使用。

降低待机功耗的做法

模块内建触控 IC，其主要功能是侦测手指是否接触到指纹传感器，当手指碰触指纹传感器的周围的铁框时，触控 IC 接口 pin 4 (ICPCK) 会输出高电平，若无接触则为低电位。

在日常的操作里，我们可以在侦测到有手指按压在指纹传感器上时再供电给指纹识别模块，如此可以降低模块待机时的功耗。我们可以使用外接的电路或微控制器(例如 Arduino 板)在侦测到 ICPCK 脚位为高电平时再供电给整个模块(透过 UART 接口 pin 4 VCC 脚)。

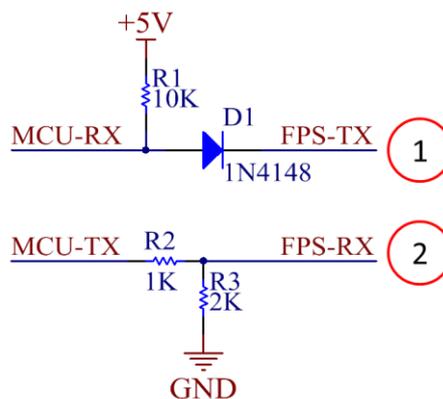
若要采用此作法来节省耗电，须注意必须保持供电给触控 IC 以便其正常工作。如果您不需要此功能，可以不使用触控 IC 接口，此接口不接线并不影响整个指纹识别模块的正常工作，但此时须保持 UART 接口 pin 4 VCC 脚持续供电(3.3V ~ 6V)的状态。

与 Arduino Uno 连接使用

此模块有丰富的指令，外部控制器可使用这些指令来操作本模块。以下介绍使用 Arduino Uno 来操作指纹识别模块 GTM-5210F32。

电平转换电路

由于 Arduino Uno 板为 5V 工作电压而指纹识别模块的 UART 电压为 3.3V，因此我们需要做 5V 与 3.3V 的逻辑电平转换，下图是一个简单的 5V 与 3.3V TTL 电平转换电路。



5V/3.3V logic Level Converter Circuit

接线

配合范例程序使用时，请依以下的接线方式。

Arduino UNO 板的接线：

Pin	功能	接线方式
4	Software serial RX	接到电平转换电路的 MCU-RX
5	Software serial TX	接到电平转换电路的 MCU-TX
6	手指按压侦测	接到指纹识别模块 Touch IC 接口的 ICPCCK (pin 4)，当手指触摸到模块传感器的铁框时，ICPCCK 引脚为 High，未接触时为 Low [注:我们提供的范例程序未使用到此讯号，可以不接此线]
5V	提供 5V 电源	接到指纹识别模块 UART 接口的 VCC (pin 4)，提供电源给指纹识别模块，同时也接到电平转换电路的+5V
3.3V	提供 3.3V 电源	接到指纹识别模块 Touch IC 接口的 VDD (pin 1)，提供电源给指纹识别模块里的 Touch IC
GND	接地	接到接到指纹识别模块 UART 接口的 GND (pin 3)，Touch IC 接口的 GND (pin 2)，同时也接到电平转换电路的 GND

指纹识别模块的接线：

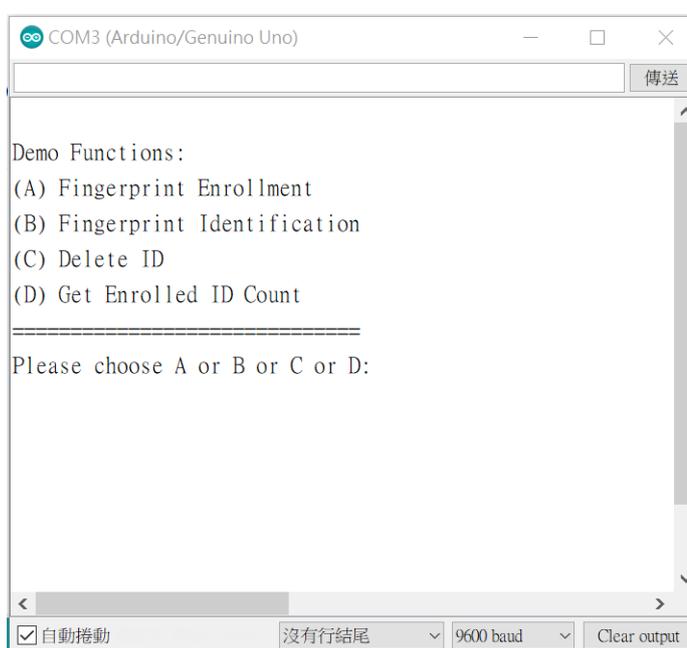
连接器	Pin	Pin Name	接线方式
UART	1	TX	接到电平转换电路的 FTP-TX
	2	RX	接到电平转换电路的 FTP-RX
	3	GND	接到 Arduino UNO 板的 GND
	4	VCC	接到 Arduino UNO 板的 5V
Touch IC	1	VDD	接到 Arduino UNO 板的 3.3V
	2	GND	接到 Arduino UNO 板的 GND
	3	ICPDA	无
	4	ICPCCK	接到 Arduino UNO 板的 pin 6 [注:我们提供的范例程序未使用到此讯号，可以不接此线]

下载并执行范例程序

网络上的开源社群已经有针对此系列指纹辨识模块的 Arduino 开源程序[\[链接\]](#),您可以参考,我们基于此开源程序开发了范例程序 FPS_GTM5210F32_demo,请在本产品产品页面的 DOCUMENTS 下载,下载后将其加入开源程序的 examples 目录内。

进入 Arduino IDE 后,打开工具 → 串行端口监控窗口 (Ctrl+Shift+M),显示初始画面如下,范例程序演示四个功能:指纹登记、指纹识别、删除 ID、查询已登记的指纹数目。

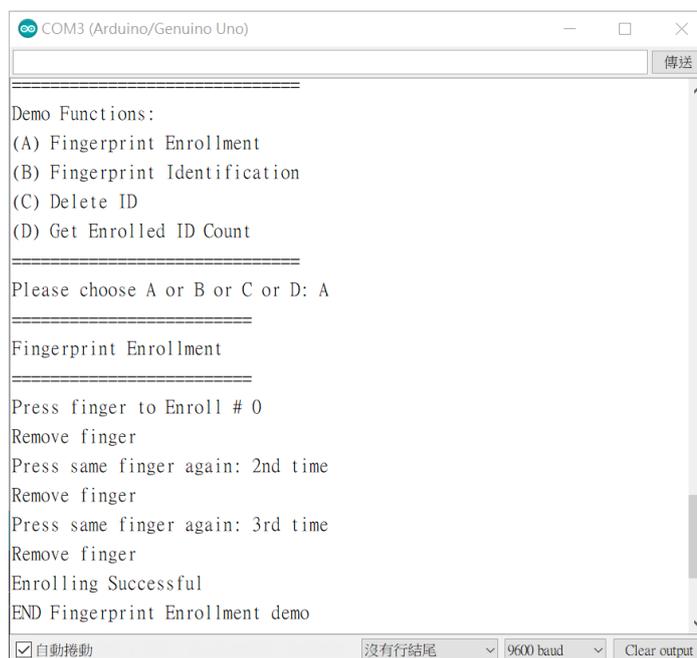
范例程序的 setup() 程序里的 UseSerialDebug 可用以设定是否显示通讯规约的封包内容,范例程序设定为不显示(false),您可自行修改为显示(true)。



指纹登记

输入 A 可执行指纹登记,范例程序会从 0 开始找到一个尚未被使用的 ID 来作为此次登记的 ID,每次登记一个指纹须按压 3 次,依照画面的提示讯息进行按压手指及移开手指即可,

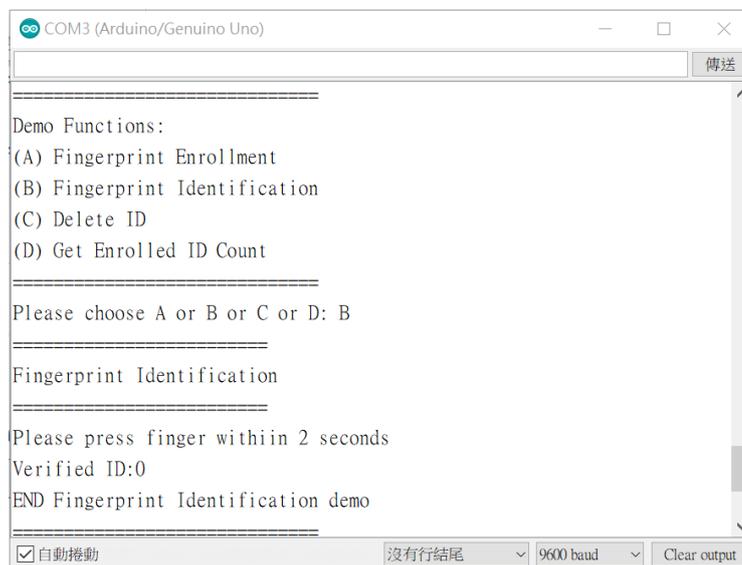
指纹登录程序主要使用指纹识别模块的 CaptureFinger 指令及 Enrollx 指令 (Enroll1、Enroll2、Enroll3)。CaptureFinger 指令的功能为取指纹,如果在执行 CaptureFinger 指令时手指未按压,则显示错误讯息 Fail to capture x finger (x 为 first、second、third)。Enrollx 指令的功能为产生指纹样板文件(template),如果指令执行失败则显示错误讯息 Enrolling Failed with error code: x; x=1 及 x=2 都是代表产出样板档失败,可能的原因是按压力道太轻,指纹不够清晰。x=3 代表指纹已经登录过了,在数据库中存在相同指纹,您可使用 Delete ID 功能来移除指纹。



```
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
```

指纹识别

输入 B 可执行指纹识别，若识别成功会显示该指纹的 ID 号码。

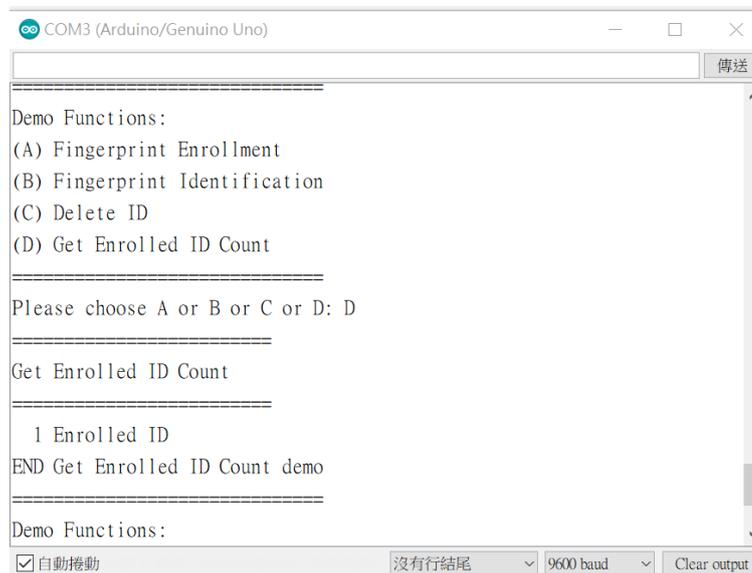


```
COM3 (Arduino/Genuino Uno)
=====
Demo Functions:
(A) Fingerprint Enrollment
(B) Fingerprint Identification
(C) Delete ID
(D) Get Enrolled ID Count
=====
Please choose A or B or C or D: B
=====
Fingerprint Identification
=====
Please press finger within 2 seconds
Verified ID:0
END Fingerprint Identification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

范例程序的设计是点亮LED后等待2秒再取指纹，您可自行修改程序调整等待时间。指纹辨识程序主要使用指纹识别模块的 CaptureFinger 指令及 Identify 指令，若在取指纹时失败，则显示错误讯息 Please press finger，未按压手指。若辨识失败则显示错误讯息 Finger not Found，原因是此指纹未登录过。

查询已登录的指纹数目

输入 D 可查询已登录的指纹数目。

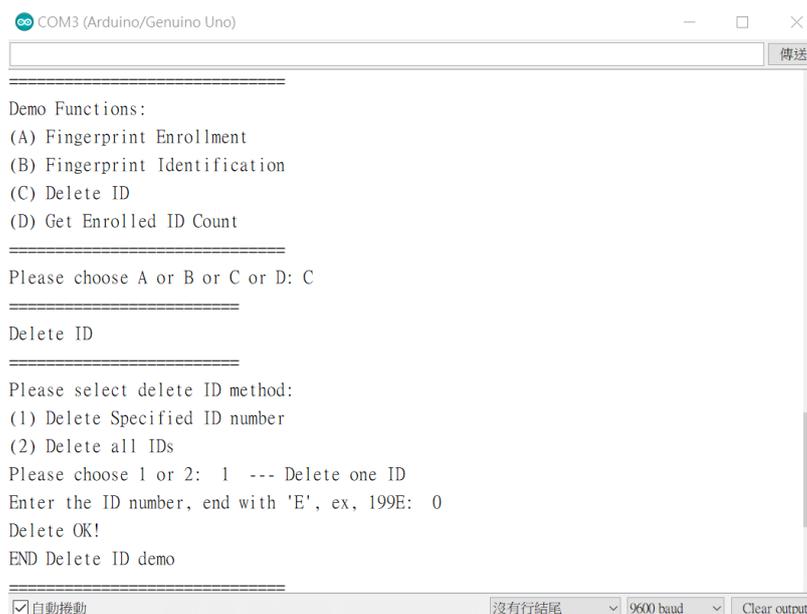


```
COM3 (Arduino/Genuino Uno)
Demo Functions:
(A) Fingerprint Enrollment
(B) Fingerprint Identification
(C) Delete ID
(D) Get Enrolled ID Count
=====
Please choose A or B or C or D: D
=====
Get Enrolled ID Count
=====
1 Enrolled ID
END Get Enrolled ID Count demo
=====
Demo Functions:
 自動捲動  沒有行結尾  9600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

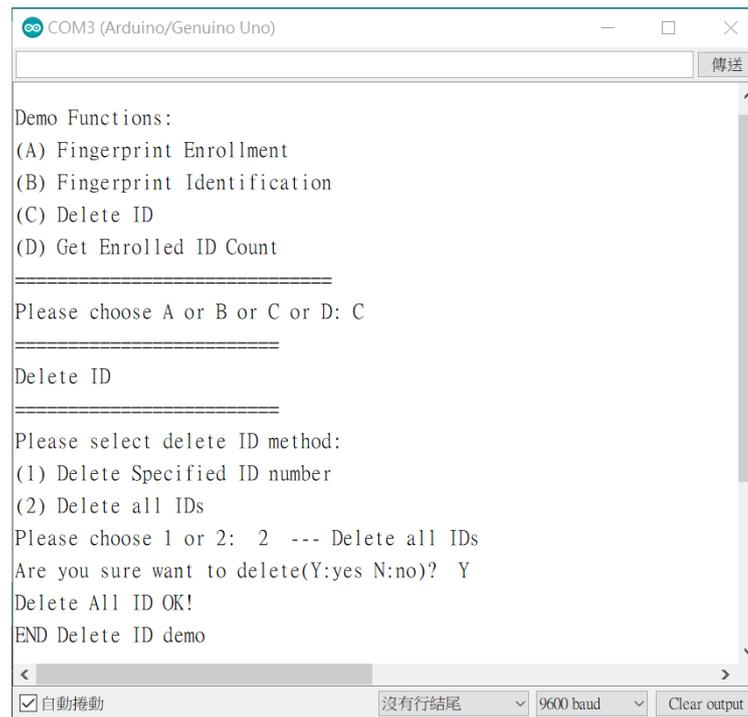
此功能主要使用指纹识别模块的 GetEnrollCount 指令，用以取得模块上的数据库目前储存的指纹数目。

删除 ID

输入 C 可执行删除 ID，可选择删除指定的 ID 或是删除所有 ID 回复到空白状态。依范例程序的设计，指定 ID 时请注意在 ID 号码后加上 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, end with 'E', ex, 199E: 0
Delete OK!
END Delete ID demo
=====
 自動捲動  沒有行結尾  9600 baud  Clear output
```



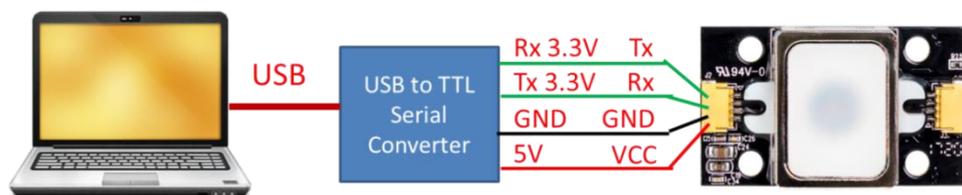
此功能主要使用指纹识别模块的 DeleteID 及 DeleteAll 指令。

使用 SDK_DEMO 软件

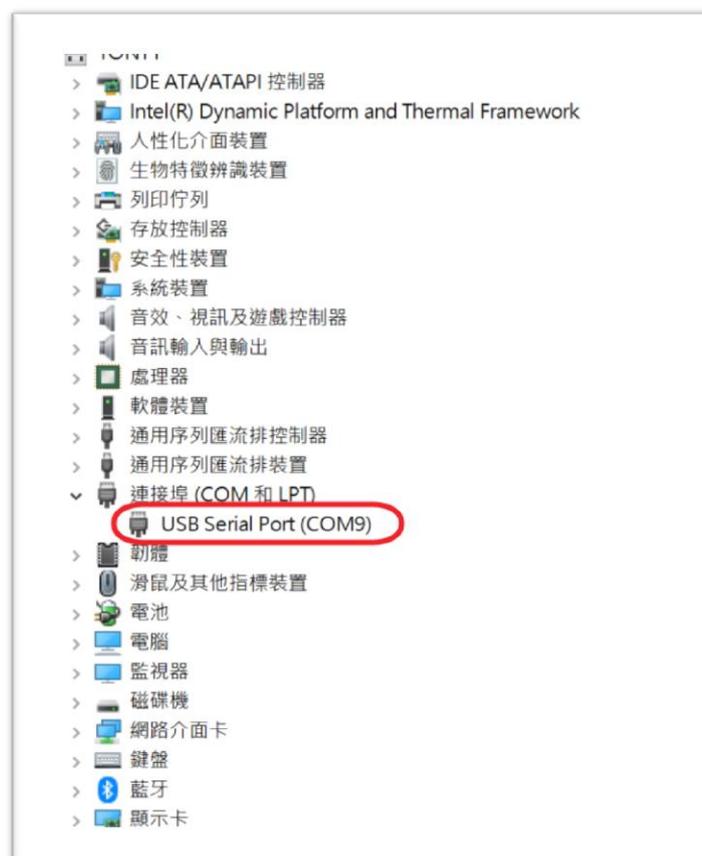
SDK_DEMO 是一个在计算机上执行的软件，它可以演示指纹识别模块 GTM-5210F32 的大部分基本功能。您只需要正确地将模块联机，不需要写程序即可开始操作指纹识别模块，SDK_DEMO 程序可以在产品页面的 DOCUMENTS 里下载。

接线

我们使用市售的 USB 转 TTL 模块来连接指纹识别模块，由于指纹识别模块的 UART 是 3.3V，所以设定 USB 转 TTL 模块工作在 3.3V，但把 5V 电源单独拉出供电给指纹识别模块(接到 UART 接口的 pin 4 VCC 脚)。如果 USB 转 TTL 模块的 5V 输出电流不足，则必须将 VCC 脚单独供电，电压范围 3.3V~6V。

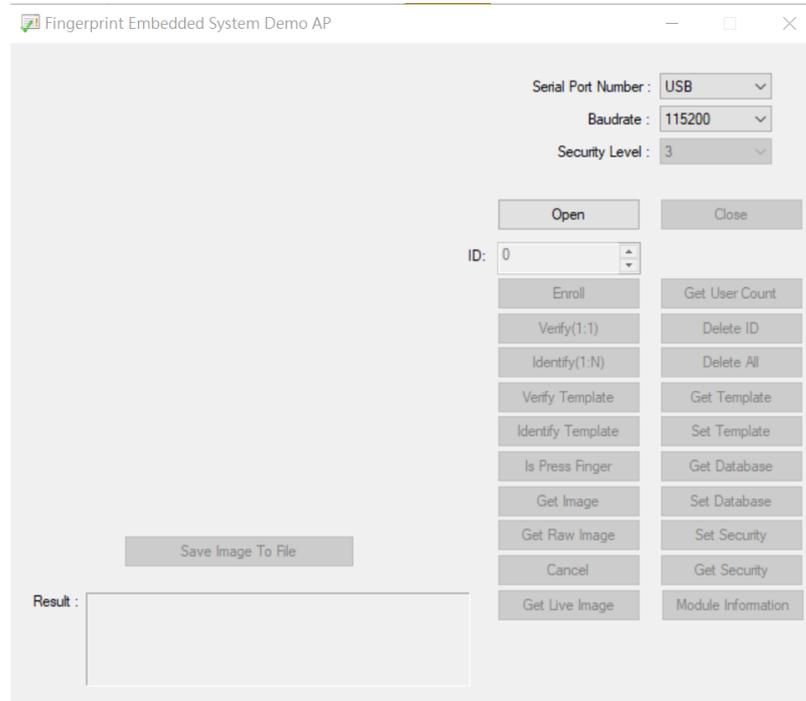


完成接线之后，在 windows 的设备管理器里可查看 USB 转 TTL 模块被指定的 COM 端口，请注意这个 COM 端口不是固定的，它会随着计算机的配置而改变。

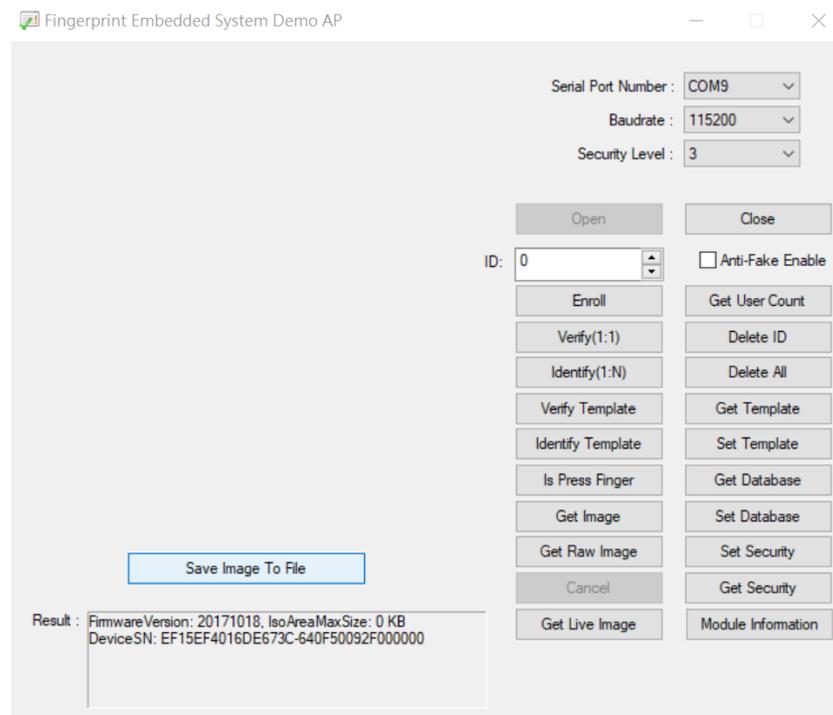


下载并执行 SDK_DEMO

启动 SDK_DEMO 后的画面如下:

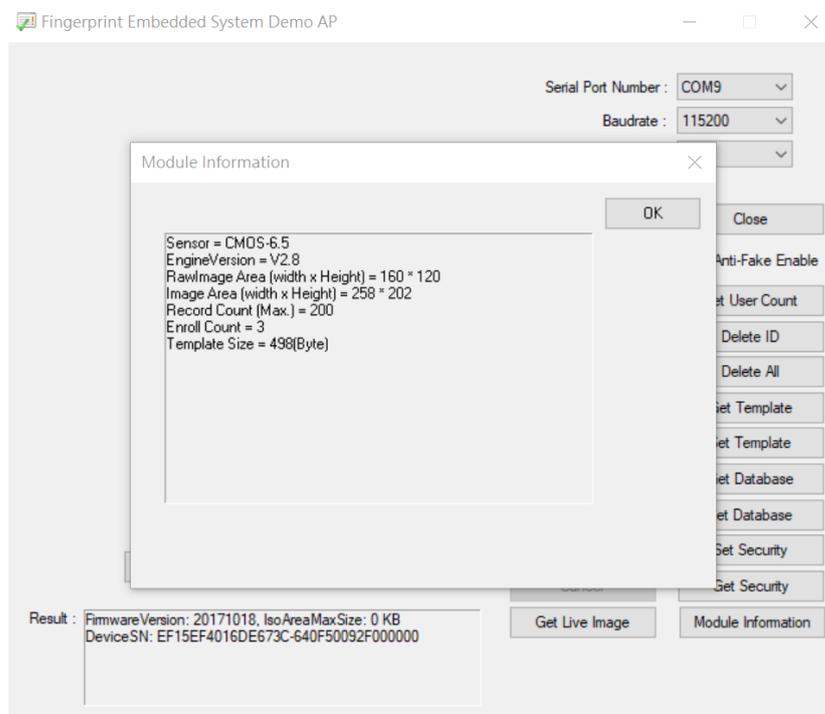


修改右上方 Serial Port Number 后再点击 Open 即可启动模块。



查看模块详细数据

您可以点击右下角的 Module Information 查看模块的详细数据。



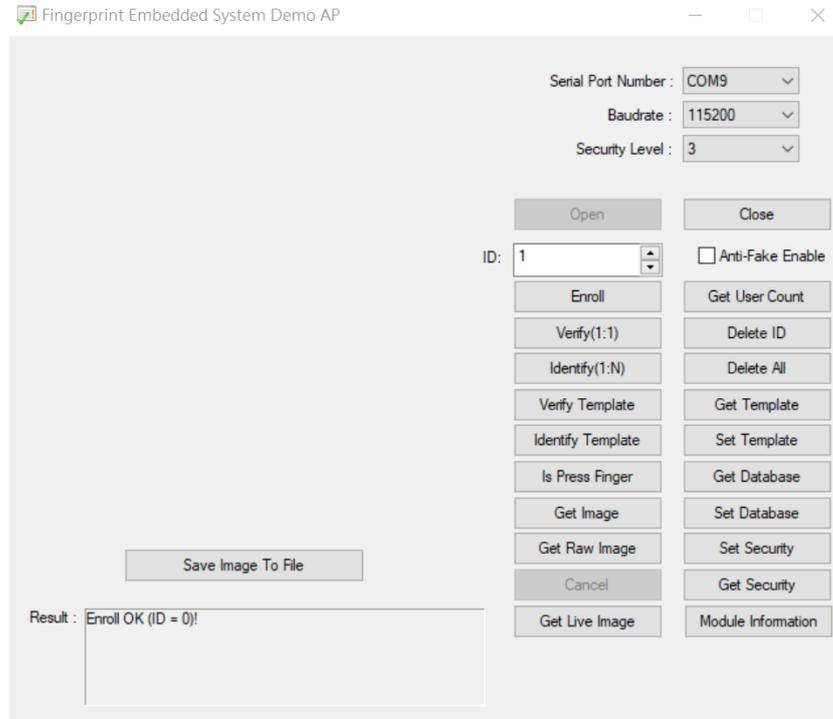
指纹登记

按压时稍微用力压紧以便于指纹采集，若按压过轻容易造成登录失败。

点击 Enroll 以登记指纹，点击前，可以更改 ID 字段里的数值以指定此次登记的指纹的 ID 编号。本模块的容量为 3000 个指纹，ID 编号由 0 到 2999，此次登记我们设定 ID 编号为 2999。

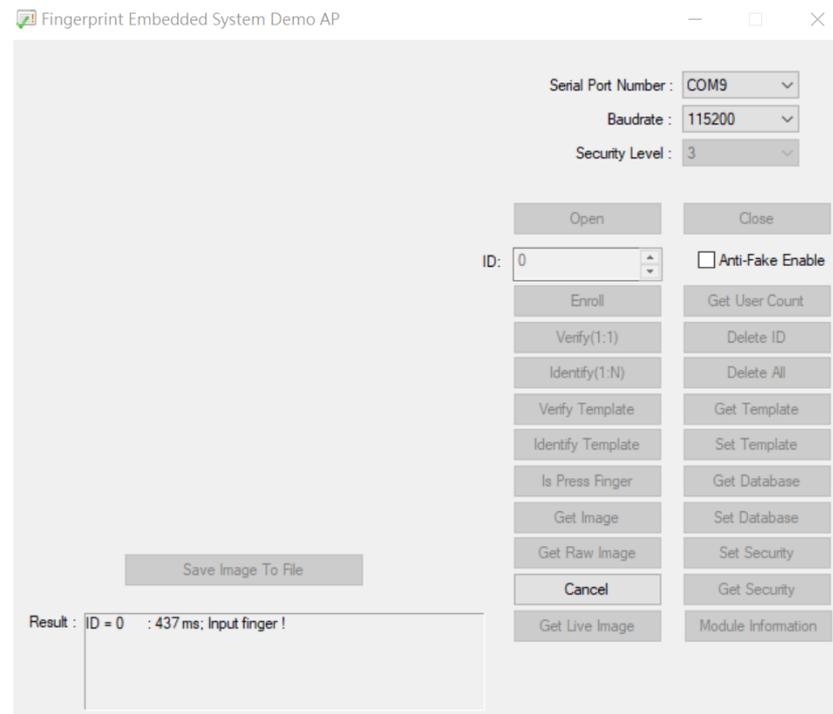
点击 Enroll 后，指纹识别模块会亮起。‘Result:’ 字段里显示”input finger 1 !’ 的信息，请您按压手指于传感器的平板上，成功登记一个指纹需要采样三次，依照 ‘Result:’ 字段的指示完成登记。

本模块支持 360 度识别，按压手指时不限于从哪个方向按压，都可以完成采样及识别。



指纹验证

Verify(1:1)的功能是比对在 ID 字段所指定的 ID 编号与按压的手指是否相同。点击 Verify(1:1)后，程序等待您按压手指，若您按压正确的手指，画面会显示 ID 及此次验证所耗用的时间。



指纹识别

Identify(1:N)的功能是比对您所按压的手指与已登记的指纹数据库，找出其 ID 编号，若您按压的手指并未登记过，画面会显示 NG。

