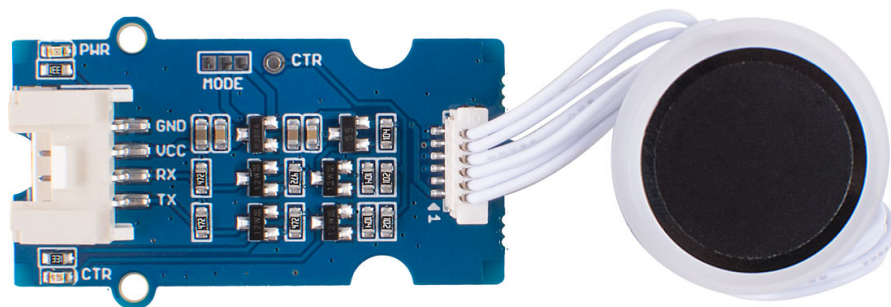# Grove - Capacitive Fingerprint Scanner/Sensor

The Grove - Capacitive Fingerprint Scanner / Sensor is based on the KCT203 Semiconductor fingerprint recognition module, including a high-performance MCU, a vertical RF push-type fingerprint sensor, and a touch sensing device. This module features many advantages such as small size, small fingerprint template, low power

consumption, high reliability, fast fingerprint recognition, etc. In addition, it is worth mentioning that there is a lovely RGB light around this module to indicate whether the fingerprint recognition is successful.

The system is equipped with a high-performance fingerprint algorithm, and the self-learning function is remarkable. After each successful fingerprint recognition, the latest challenge feature values can be integrated into the fingerprint database to continuously improve the fingerprint features, making the experience better.

We add the power level shift circuit on the Grove Driver board so that this module can work with both 3.3V and 5V systems. And with the help of the Grove UART connector and the Arduino library we offered, you can build your own Arduino fingerprint sensor/scanner easily.

We have already released the Grove optical fingerprint sensor, this time we bring you the Grove - Capacitive Fingerprint Scanner / Sensor. So what's the difference? Well, let's show you the merits and demerits of both sensors so that you can choose the best one to meet your needs.

Product	Advantage	Disadvantage
Grove - Capacitive Fingerprint Scanner / Sensor [https://www.seeedstudio.com/Grove-Capacitive-Fingerprint-Scanner-p-4363.html]	Can identify biological characteristics, only identify living body, high safety, small size, low power consumption, high accuracy	Poor abrasion resistance, easily susceptible to sweat, stains, finger wear, etc.
Grove - Optical Fingerprint Sensor [https://www.seeedstudio.com/Grove-Fingerprint-Sensor.html]	Strong abrasion resistance, good environmental adaptability, and good stability	Large size, high power consumption, and relatively low accuracy, can't identify the living body, low security

[https://www.seeedstudio.com/Grove-Capacitive-Fingerprint-Scanner-p-4363.html]

Features

- Built-in 2KByte storage: support up to 100 fingerprints

- Powerful self-learning function: the more you use, the more accurate the recognition

- Selectable security level

- Small size, low power consumption,10uA for standby mode

Applications

- Fingerprint lock devices: door locks, safes, steering wheel locks, padlocks, gun locks, etc.

- Fingerprint sign-in, access control system

Specification

Parameter	Value
CPU	GD32
Fingerprint Template Storage	Max. 100
Connector	Grove UART
Sensor Resolution	508 DPI
Sensor Pixel	160*160
False Rejection Rate	<1%
False Acceptance Rate	<0.005%
Match Response Time(1:N Mode)	<350ms
Match Response Time(1:1 Mode)	<7ms
Sensor Size	Φ14.9mm
Frame Size	Φ19mm
Power Consumption	Full speed: ≤40 mA; Sleep: ≤12uA
Operating Voltage	3.3V/5V
Operating Temperature	-20~70℃
ESD Protection	Non-contact 15KV, contact 8KV

Pin Out
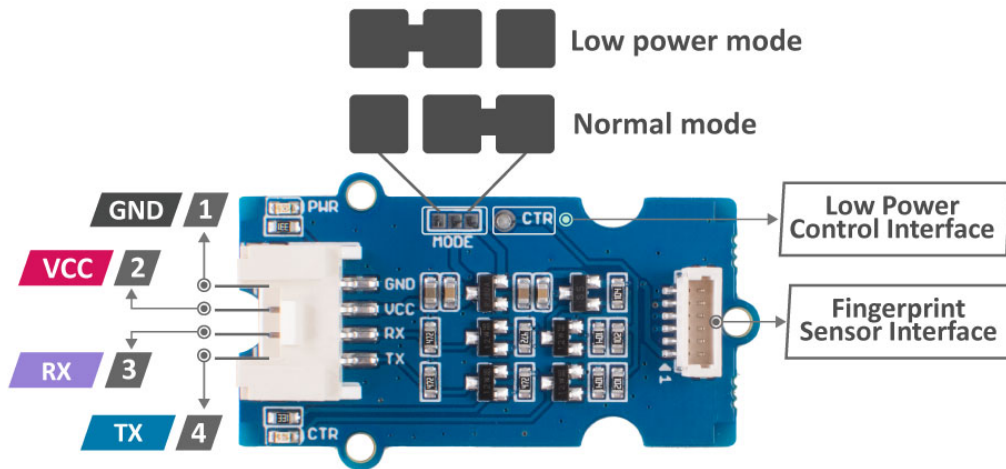
1 : Connected to the system GND

2 : Power supply from grove 5V/3.3V

3 : RX pin of UART

4 : TX pin of UART

CTR : In normal mode, the CTR signal is disabled;

in low power mode, the CTR signal is enabled,

and the signal is pulled to ground to enter sleep mode

Platforms Supported

Arduino	Raspberry Pi		

> ⚠️ **Caution**
>
> The platforms mentioned above as supported is/are an indication of the module's software or theoritical compatibility. We only provide software library or code examples for Arduino platform in most cases. It is not possible to provide software library / demo code for all possible MCU platforms. Hence, users have to write their own software library.

Getting Started

Play With Arduino

Hardware

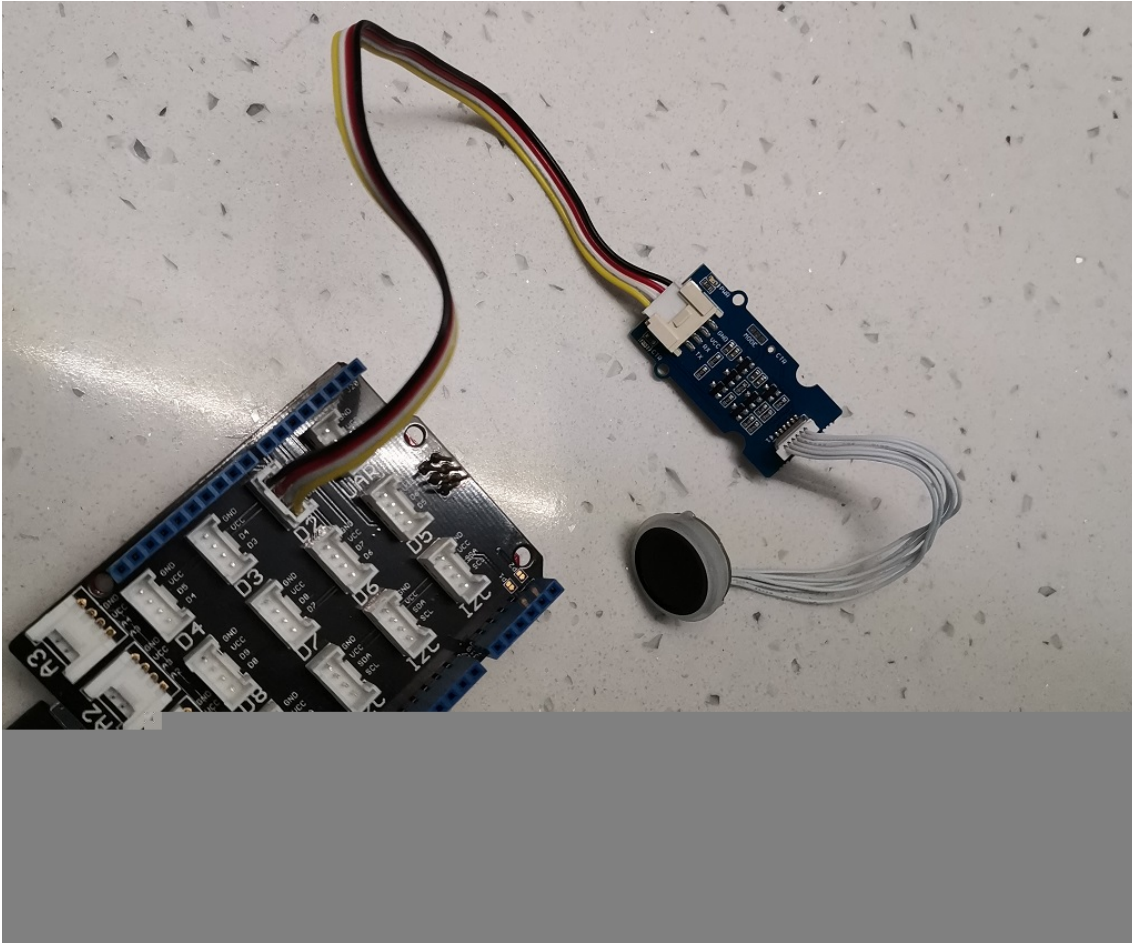| Seeeduino V4.2	Base Shield
Get One Now [https://www.seeedstudio.com/Seeeduino-V4.2-p-2517.html]	Get One Now [https://www.seeedstudio.com/Base-Shield-V2-p-1378.html]

> ✏️ **Note**
>
> **1** Please plug the USB cable gently, otherwise you may damage the port. Please use the USB cable with 4 wires inside, the 2 wires cable can't transfer data. If you are not sure about the wire you have, you can click here [https://www.seeedstudio.com/Micro-USB-Cable-48cm-p-1475.html] to buy
>
> **2** Each Grove module comes with a Grove cable when you buy. In case you lose the Grove cable, you can click here [https://www.seeedstudio.com/Grove-Universal-4-Pin-Buckled-20cm-Cable-%285-PCs-pack%29-p-936.html] to buy.

- **Step 1.** Connect Grove - Capacitive Fingerprint Scanner/Sensor to port **D2** of Grove-Base Shield.

- **Step 2.** Plug Grove - Base Shield into Seeeduino.

- **Step 3.** Connect Seeeduino to PC via a USB cable.

> ✏️ **Note**
>
> If we don't have Grove Base Shield, We also can directly connect this module to Seeeduino as below.

Seeeduino	Grove Cable	Grove - Capacitive Fingerprint Scanner/Sensor
GND	Black	GND
5V or 3.3V	Red	VCC
D3	White	RX
D2	Yellow	TX

Software

> ⚠️ **Attention**
> If this is the first time you work with Arduino, we strongly recommend you to see Getting Started with Arduino [https://wiki.seeedstudio.com/Getting_Started_with_Arduino/] before the start.

- **Step 1.** Download the Seeed_Arduino_KCT202 [https://github.com/Seeed-Studio/Seeed_Arduino_KCT202] Library from Github.

- **Step 2.** Refer to How to install library [https://wiki.seeedstudio.com/How_to_install_Arduino_Library] to install library for Arduino.

- **Step 3.** Restart the Arduino IDE. Open the example, you can open it in the following three ways：

   a. Open it directly in the Arduino IDE via the path: **File** → **Examples** → **Seeed_Arduino_KCT202** → **full_process**.

b. Or, you can just click the icon in upper right corner of the code block to copy the following code into a new sketch in the Arduino IDE.

```
1   /*
2
3     Copyright (c) 2019 Seeed Technology Co., Ltd.
4     Website    : www.seeed.cc
5     Author     : downey
6     Create Time: Jun 2019
7     Change Log :
8
9     The MIT License (MIT)
10
11    Permission is hereby granted, free of charge, to any
12    of this software and associated documentation files
13    in the Software without restriction, including witho
14    to use, copy, modify, merge, publish, distribute, su
15    copies of the Software, and to permit persons to whom
16    furnished to do so, subject to the following conditi
17
18    The above copyright notice and this permission notic
```

```cpp
19       all copies or substantial portions of the Software.
20
21       THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY O
22       IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES
23       FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT
24       AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM,
25       LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR
26       OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE
27       THE SOFTWARE.
28  */
29  #include "ATSerial.h"
30  #include "Protocol.h"
31  #include "KCT202.h"
32
33
34  #if defined(ARDUINO_ARCH_AVR)
35      #define debug  Serial
36      SoftwareSerial uart(2, 3);
37      FingerPrint_KCT202<SoftwareSerial, HardwareSerial> k
38  #elif defined(ARDUINO_ARCH_SAM)
39      #define debug  SerialUSB
40      #define uart Serial
41      FingerPrint_KCT202<Uart, Serial_> kct202;
42  #elif defined(ARDUINO_ARCH_SAMD)
43      #define debug  SerialUSB
44      #define uart Serial1
45      FingerPrint_KCT202<Uart, Serial_> kct202;
46  #else
47      #define debug  Serial
48      SoftwareSerial uart(2, 3);
49      FingerPrint_KCT202<SoftwareSerial, HardwareSerial> k
50  #endif
51
52
53
54
55
56  Protocol_oprt oprt;
57  uint8_t err_code = 0;
58  uint8_t param[10];
59  uint32_t param_len;
```

```
60
61  void setup(void) {
62      debug.begin(115200);
63      kct202.begin(uart, debug);
64  }
65
66  uint16_t finger_num;
67  void loop() {
68      //The first param is the finger-print ID to check.
69      //if set 0xffff,indicates that search for all the fi
70      kct202.autoVerifyFingerPrint(CHECK_ALL_FINGER_TEMP,
71                                   LED_OFF_AFTER_GET_GRAGH
72      debug.println(" ");
73      debug.println("Please put your finger on the touchpa
74      debug.println("To verify your finger print.");
75      debug.println(" ");
76      debug.println(" ");
77      debug.println(" ");
78
79      if (0 == kct202.getVerifyResponAndparse(finger_num))
80          debug.println("Verify ok!");
81          debug.print("Your finger temp id = ");
82          debug.println(finger_num, HEX);
83      }
84      delay(2000);
85  }
```

> ⚠ **Attention**
> The library file may be updated. This code may not be applicable to the
> updated library file, so we recommend that you use the first way.

- **Step 4.** Upload the demo. If you do not know how to upload the
  code, please check How to upload code
  [https://wiki.seeedstudio.com/Upload_Code/].

- **Step 5.** Open the **Serial Monitor** of Arduino IDE by click **Tool->
  Serial Monitor**. Or tap the `Ctrl` + `Shift` + `M` key at the same

time. Set the baud rate to **115200**.

- **Step 6.** Please put your finger on the touchpad.

> ✓ **Success**
> If every thing goes well, when you open the Serial Monitor, it may show as below:

```
 1   Usage:
 2       Please put your finger on the touchpad
 3       Repeat for 4 times or pressed continuously
 4
 5
 6
 7   Uart<=== : EF.1.FF.FF.FF.FF.7.0.5.0.0.0.0.C.
 8
 9   ****************************************************
10   Please put your finger on the touchpad
11   ****************************************************
12   Uart<=== : EF.1.FF.FF.FF.FF.7.0.5.0.1.1.0.E.
13
14   ****************************************************
15   Please put your finger on the touchpad
16   ****************************************************
17   Uart<=== : EF.1.FF.FF.FF.FF.7.0.5.0.2.1.0.F.
18
19   ****************************************************
20   Please put your finger on the touchpad
21   ****************************************************
22   Uart<=== : EF.1.FF.FF.FF.FF.7.0.5.0.1.2.0.F.
23
24   ****************************************************
25   Please put your finger on the touchpad
26   ****************************************************
27   Uart<=== : EF.1.FF.FF.FF.FF.7.0.5.0.2.2.0.10.
28
29   ****************************************************
30   Please put your finger on the touchpad
31   ****************************************************
```

```
32  Uart<=== : EF.1.FF.FF.FF.FF.7.0.5.0.1.3.0.10.
33
34  *****************************************************
35  Please put your finger on the touchpad
36  *****************************************************
37  Uart<=== : EF.1.FF.FF.FF.FF.7.0.5.0.2.3.0.11.
38
39  *****************************************************
40  Please put your finger on the touchpad
41  *****************************************************
42  Uart<=== : EF.1.FF.FF.FF.FF.7.0.5.0.1.4.0.11.
43
44  *****************************************************
45  Please put your finger on the touchpad
46  *****************************************************
47  Uart<=== : EF.1.FF.FF.FF.FF.7.0.5.0.2.4.0.12.EF.1.FF.FF.
48
49  Register ok!
50  Uart===> : EF.1.FF.FF.FF.FF.1.0.8.32.0.FF.FF.0.7.2.40.
51
52
53  Please put your finger on the touchpad.
54  To verify your finger print.
55
56
57
58  Uart<=== : EF.1.FF.FF.FF.FF.7.0.8.0.0.0.0.0.0.0.F.
59
60  Uart<=== : EF.1.FF.FF.FF.FF.7.0.8.0.1.0.0.0.0.0.10.
61
62  Uart<=== : EF.1.FF.FF.FF.FF.7.0.8.0.5.0.1.0.64.0.79.
63
64  Verify ok!
65  Your finger temp id = 1
66  Uart===> : EF.1.FF.FF.FF.FF.1.0.3.D.0.11.
67
68  Uart<=== : EF.1.FF.FF.FF.FF.7.0.3.0.0.A.
69
70  Operation succed.
71  Delete ok!
```

Schematic Online Viewer

Resources

- **[Zip]** Grove - Capacitive Fingerprint Scanner/Sensor eagle file [https://files.seeedstudio.com/wiki/Grove-Capacitive-Fingerprint-Scanner/res/Grove-Capacitive-Fingerprint-Scanner_SCH&PCB.zip]

- **[Zip]** Grove - Capacitive Fingerprint Scanner/Sensor code [https://github.com/Seeed-Studio/Seeed_Arduino_KCT202/archive/master.zip]

Tech Support

Please do not hesitate to submit the issue into our forum [https://forum.seeedstudio.com/]

[https://www.seeedstudio.com/act-4.html?
utm_source=wiki&utm_medium=wikibanner&utm_campaign=newpr
oducts]