Capacitive Fingerprint Reader

OVERVIEW

The Capacitive Fingerprint Reader is a standard fingerprinting module designed for secondary development, allows fast and stable fingerprint verification.

Based on a high-speed digital processor STM32F105R8, combined with high-security commercial fingerprinting algorithm, and advanced semiconductor sensor, the Capacitive Fingerprint Reader becomes a simple but smart integration module, provides functionalities like fingerprint enrolling, image processing, feature finding, template generating and storing, fingerprint matching and searching, etc.

FEATURES

- Easy to use with some simple commands, you don’t have to know any fingerprint technology, or the module inter structure/calculation
- Commercial fingerprinting algorithm, stable performance and fast verification.
- Sensitive detection, just touch the collecting window lightly, without pressing.
- Allows to freely input/output fingerprint images, fingerprint feature file and other fingerprinting actions
- Dual communication, UART or USB
- Processor will enter sleep and wake up when detecting, achieves low power consumption
- Comes with rich related resources (software on PC, development SDK, tools, and examples for Raspberry Pi/Arduino/STM32)

APPLICATIONS

Fingerprint lock, fingerprint safe deposit box, access control system, clock in and out and so on.
Capacitive Fingerprint Reader

CONTENT

Overview ................................................................................................................................. 1
Features ................................................................................................................................. 1
Applications ............................................................................................................................ 1
How to use ............................................................................................................................ 4
Specifications ......................................................................................................................... 4
Interface ............................................................................................................................... 4
Hardware connection ............................................................................................................ 4
Operating Instructions .......................................................................................................... 5
Dimension of sensor ............................................................................................................. 5
UART port ............................................................................................................................. 6
Examples of Operating .......................................................................................................... 8
Additional operations ........................................................................................................... 11
USB control description ....................................................................................................... 14
Examples ................................................................................................................................ 18
XNUCLEO-F103RB .................................................................................................................. 18
    Hardware connection ........................................................................................................... 18
    Programming ...................................................................................................................... 19
    Commands description ...................................................................................................... 20
UNO PLUS ................................................................................................................................ 20
    Hardware connection ........................................................................................................... 20
    Programming ...................................................................................................................... 21
Raspberry Pi ............................................................................................................................ 21
    Hardware connection ........................................................................................................... 21
    Raspberry Pi serial port configuration .............................................................................. 22
    Python ............................................................................................................................... 22
    WiringPi .............................................................................................................................. 22

Date: 2017 Nov. 13
# HOW TO USE

## SPECIFICATIONS

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Processor (CPU)</td>
<td>STM32F105</td>
</tr>
<tr>
<td>Sensor</td>
<td>Semiconductor (capacitive)</td>
</tr>
<tr>
<td>Module dimension (mm)</td>
<td>45x30</td>
</tr>
<tr>
<td>Sensor dimension (mm)</td>
<td>33.4x20.4</td>
</tr>
<tr>
<td>Image</td>
<td>DPI508</td>
</tr>
<tr>
<td>Image resolution</td>
<td>192x256</td>
</tr>
<tr>
<td>Sensing area (mm)</td>
<td>9.6x12.8</td>
</tr>
<tr>
<td>Fingerprint capacity</td>
<td>1000 (expandable)</td>
</tr>
<tr>
<td>Security level</td>
<td>0-9 low to high security, 5 by default</td>
</tr>
<tr>
<td>Matching time</td>
<td>1s</td>
</tr>
<tr>
<td>False acceptance rate</td>
<td>0.0001%</td>
</tr>
<tr>
<td>False rejection rate</td>
<td>0.01%</td>
</tr>
<tr>
<td>Dynamic current</td>
<td>&lt;40mA</td>
</tr>
<tr>
<td>Operating voltage</td>
<td>3.3V-5V</td>
</tr>
<tr>
<td>Communication</td>
<td>UART or USB</td>
</tr>
<tr>
<td>Anti-electrostatic</td>
<td>aerial discharge: ±15KV</td>
</tr>
<tr>
<td></td>
<td>contact discharge: ±8KV</td>
</tr>
</tbody>
</table>

Note: Replace the memory chip with W25Q32, the capacity could be expanded to 4000

## INTERFACE

VCC  3.3V or 5V
GND  Ground
TXD (serial output of fingerprint module)  RXD (serial input of PC or MCU)
RXD (serial input of fingerprint module)  TXD (serial output of PC or MCU)
WAKE (collector wake up, connection is not a must)  IO pin
RST (reset, connection is not a must)  IO pin

### HARDWARE CONNECTION

Date: 2017 Nov. 13
Connecting the sensor and control board with incongruous cable as below:

After receiving the Fingerprint Reader, users can test the module firstly by test software. Users could use the protocols below for further developing after mastering enough information of this fingerprint reader.

This Capacitive Fingerprint Reader communicate via UART or USB interface on board.

**OPERATING INSTRUCTIONS**

This module uses high-precision components, while collecting fingerprint:

You only need to touch the collecting window gently, not need to press it hard, fingerprint could be identified quickly.

- **Correct Fingerprint**

- **Wrong Fingerprint**

**DIMENSION OF SENSOR**
1. Testing the module by using CP2102 USB UART Board. (Should buy separately). If users use other UART to USB module, the operation is same.). Install CP2102 driver

2. Connecting as below: (note that RXD and TXD are connected crossly):
3. Connecting the CP2102 board to the USB port of PC. Open Devices Manager to check the appropriate COM port:

4. Download testing software: Capacitive-Fingerprint-Reader (UART), you need to register the control at first use:

   Registering MSCOMM32.OCX, if there is error prompt popped out that: The module "mscomm32.ocx" was loaded but the call to DllRegisterServer failed with error code 0x8002801c.

   You can solve this problem with the following methods:

   For 64-bit OS, copy the MSCOMM32.OCX file to the directory of C:\Windows\SysWOW64, and then run the register script. If it is still doesn’t work, you need to copy the MSCOMM32.OCX to the directory of C:\Windows\System32, and run the register script. (It
may require the administrator privilege to register the control successfully in this method.

Open the UART Fingerprint Reader, you will see the operation interface as the below figure shows if the module works properly. (The default port of the USB to serial interface was set to COM1, if your COM1 isn’t in use, there will be a prompt saying that *Comm port error!* In this case, just click OK then choose the corresponding COM port in use.

**EXAMPLES OF OPERATING**
1. Configure the COM port: select the tab **Setting**, and choose the correct COM port in the option **Communication port selection** (Default baud rate: 19200).

![Setting tab with COM port selection](image)

2. Check the communication: select the tab **Test**, and choose the option **Get the total number of user** in the area of **Basic operation**. Then, click the button **Send**. You will see the relative information displayed in the box of **Display returned data** (e.g. it prompts "用户总数：0" which means "Total number is users: 0"), of the communication between the module and your PC is built up successfully. (In case of no information shown, please turn back to the previous steps to check carefully whether there are any mistakes.)

![Test tab with communication test](image)
3. **User fingerprint input**: In group **Basic Operation**, select the option **User fingerprint input**, and check the options **Acquire twice** and **ID auto-increment by 1**. Set the parameters P1, P2, and P3 in the group **Parameter selection** to 0, 1, 1, respectively. (You can set the parameters to other numbers as well, if only there is no ID duplicates) Click the button **Send**. When you hear a 'beep' from the module and find the fingerprint sensing area light up, you can put your finger on the sensing area for scanning. After the second scanning finished, it will prompt “用户录入成功” which means “Input Fingerprint Successfully”. Please take a notice that do not check the option Upload feature. This option may upload the characteristics to the PC other than fingerprint reader, which will cause fingerprint marching failure and no increase in the total number of user, though you may get the information of ‘fingerprint input success!’ after a new fingerprint is added.

4. **Fingerprint comparison**: select the option **1:N comparison** in the area of **Basic operation**, and click the button **Send**. When you hear a 'beep' from the module and find the fingerprint sensing area light up, you can put your finger on the sensing area for scanning. If the fingerprint scanned is already stored into the module, the module will present the information “n 号用户比对成功” which means “No. n user matched”. Or else, you may get
the information "No match".

ADDITIONAL OPERATIONS

Note: the following operations involve the baud rate modifications, if you have changed the baud rate, and you can’t already make connection next time. Please check the baud rate first, whether it is normal!

1. Baud rate change: for example, hang change the baud rate from 19200 to 115200:

Note: in this case the baud rate has been changed to 115200, please notice before you use it next time that the baud rate is configured to 115200, otherwise the communication will be failed.
If you've forgotten the baud rate, you can use this method to define it:

Start the software **Serial Debugging Assistant**, and select the corresponding COM port. Then, open the COM port and check the options HEX display and HEX transmission. Input a string off data, such as '5 09 00 00 00 09 09 F5', into the string input box, and click the button Send. If the baud rate setting is correct, the receive box will show the relative data, such as 'F5 09 00 01 00 00 08 F5' in here. For a wrong baud rate setting, there will be nothing displayed. In this way, you can get back the correct baud rate by trying different baud rate setting.

2. Obtaining of fingerprint image: select the option Acquire fingerprint image in the area of Basic operation, and click the button Send. When you hear a 'beep' from the module and
find the fingerprint sensing area light up, you can put your finger on the sensing area for scanning. It may take 6 seconds before displaying the fingerprint image acquired on the PC.

Note: If the module cannot acquire any image or the image acquired is imperfect, you should reduce the baud rate to make sure the quality of the image. It is recommended to use the baud rate of 9600 for fingerprint image acquirement.

3. Sleep mode: select the option Sleep in the area of Basic operation, and click the button Send. Then, you will get the information of "模块进入休眠状态,不再响应任何命令" which means "Module enters sleep mode and no command will be responded". The only way to
wake up a sleeping module is to power it up once again.

USB CONTROL DESCRIPTION

1. Plug the Fingerprint Reader. It simulates the CD driver interface, thus no need to install any driver. Plug directly and the module can be used immediately.

The following picture shows the case if there is no access.

The following picture shows the case of access (devices is displayed as CD drive)
2 Start testing software
   2.1 Download testing software: Capacitive-Fingerprint-Reader_(UART), extract and open "\D5ScannerS77Demo\Release\D5ScannerS77Demo.exe"

   2.2 Define device, Click button “Enum Device” to enumerate all of Fingerprint Readers connected to PC this moment, then select appropriate device from the box as below:
2.3 Open the device: Click the button “Open Device”, it will prompt that “Open Device OK” or “Open Device Fail”
2.4 Press the Fingerprint Reader by finger, ever fingerprint scanned, it will sound “beep”.

2.5 Save image by pushing “Save Image” button, select *.bmp file extension
2.6 Save characteristics by pushing “Save Feature” button, select *.fea extension

![Image of software interface for saving fingerprint characteristics]

**EXAMPLES**

In this chapter, we use XNUCLEO-F103RB, UNO PLUS and Raspberry Pi 3B to show you how to use the fingerprint reader. (using UART serial port).

**XNUCLEO-F103RB**

- Based on STM32F103RBT6
- Based on HAL library, it can be ported to any other STM chip by STM32CubeMX
- Can be compiled in Keil v5

**HARDWARE CONNECTION**

<table>
<thead>
<tr>
<th>Capacitive Fingerprint Reader</th>
<th>XNUCLEO-F103RB</th>
</tr>
</thead>
<tbody>
<tr>
<td>VCC</td>
<td>3.3V</td>
</tr>
<tr>
<td>GND</td>
<td>GND</td>
</tr>
</tbody>
</table>

Date: 2017 Nov. 13
Capacitive Fingerprint Reader

<table>
<thead>
<tr>
<th>TXD</th>
<th>D2 (PA10)</th>
</tr>
</thead>
<tbody>
<tr>
<td>RXD</td>
<td>D8 (PA9)</td>
</tr>
<tr>
<td>WAKE</td>
<td>D3 (PB3)</td>
</tr>
<tr>
<td>RST</td>
<td>D4 (PB5)</td>
</tr>
</tbody>
</table>

**PROGRAMMING**

1. After connecting the module to development board, plug the ST-Link programmer, open file *Capacitive Fingerprint Reader.uvprojx* which is under MDK-ARM directory. Click "Build" to compile the project, then click "Download" button, the project will be flashed into the chip.

2. Connecting the development board to PC by USB cable, open the serial port monitoring application, select right serial port number, and setup following settings:
   - baud rate: 115200; bits: 8; stop bit: 1;check bit: none; control flow: none

3. Press button "Reset" on the board, you can see operation tips are printed in the serial port assistant:

```
*************** Waveshare Capacitive Fingerprint Reader Test ***************
Compare level: 5. (can be set to 0-9, the bigger, the stricter)
Number of fingerprints already available: 1
Use the serial port to send the commands to operate the module:
CMD1 : Query the number of existing fingerprints
CMD2 : Add fingerprint (Each entry needs to be read two times: "beep", put the finger on sensor, "beep", put up, "beep", put on again)
CMD3 : Fingerprint matching (Send the command, put your finger on sensor after "beep". Each time you send a command, module waits and matches once)
CMD4 : Clear fingerprints
CMD5 : Switch to sleep mode, you can use the finger to wake up function. (In this state, only CMD6 is valid. When a finger is placed on the sensor, the module is waken up and the finger is matched, without sending commands to match each time. The CMD6 can be used to wake up)
CMD6 : Wake up and make all commands valid
*************** Waveshare Capacitive Fingerprint Reader Test ***************
```

Note: if the serial assistant always returns the error message "***ERROR***", please ensure that the module is powered by 3.3V or 5V, the serial wired connection is correct and the module baud rate is 19200 by default. Then power off, and power on it again.

4. The message above informs that the current matching level of module is 5, the current comparison level is 5 (can be set to 0-9, the bigger, the stricter). And there is one fingerprint saved in the module. Users can send commands CMD1-CMD6 by serial assistant to operate the module.

**Note**: Do not check the option "发送新行" (send new line) and "HEX 发送" (HEX to send).
after sending commands, please pay attention to the operation tips responded!

<table>
<thead>
<tr>
<th>COMMANDS DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>CMD1: Query the number of existing fingerprints.</td>
</tr>
<tr>
<td>CMD2: Add fingerprint (Each entry should be read twice, “beep”, put your finger on the sensor. “beep”, take away, “beep”, put your finger again). Please put the finger on the center of the sensor.</td>
</tr>
<tr>
<td>CMD3: Fingerprint match (Send command, put your fingerprint after is sound “beep”. Each time you send a command, the module waits and matches one). Please put the finger on the center of the sensor.</td>
</tr>
<tr>
<td>CMD4: Clear database of fingerprints.</td>
</tr>
<tr>
<td>CMD5: Switch to sleep mode. You can use the finger automatic wake-up function (after sending this command, only the CMD6 is available. When a finger being putted on the sensor the module wakes up and matches the fingerprint. Needn’t send CMD3 for matching every time. Send CMD6 to exit the sleep mode).</td>
</tr>
<tr>
<td>CMD6: Switch to work mode. Wake up and make all commands valued</td>
</tr>
</tbody>
</table>

**UNO PLUS**

**HARDWQRE CONNECTION**

<table>
<thead>
<tr>
<th>Capacitive Fingerprint Reader</th>
<th>Arduino UNO PLUS</th>
</tr>
</thead>
<tbody>
<tr>
<td>VCC</td>
<td>3.3V or 5V</td>
</tr>
<tr>
<td>GND</td>
<td>GND</td>
</tr>
</tbody>
</table>
Capacitive Fingerprint Reader

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>TXD</td>
<td>D10</td>
</tr>
<tr>
<td>RXD</td>
<td>D11</td>
</tr>
<tr>
<td>WAKE</td>
<td>D8</td>
</tr>
<tr>
<td>RST</td>
<td>D9</td>
</tr>
</tbody>
</table>

Note: since the Arduino UNO PLUS has only one hardware serial port and this port is used to connect to the PC for downloading and as a debugging tool, so here we use D10, D11 to simulate a software serial port for communication with Capacitive Fingerprint Reader.

PROGRAMMING

1. Copy the Arduino\Capacitive_Fingerprint_Reader folder from package to Arduino installation directory Arduino\libraries. Open project at File → Examples → Capacitive_Fingerprint_Reader → Finger_Test, compile and download.

2. Click Tools → Port, select serial port number of Arduino development board, Open the serial monitor and configure as No line ending, 115200 baud. Press the button “Reset” on the board, you can easily see in the serial port assistant following:

Note: if the serial assistant always returns the error message “***ERROR***”, please ensure that the module is powered by 3.3V or 5V, the serial wired connection is correct and the module baud rate is 19200 by default. Then power off, and power on it again. The specification of operation methods, commands and warning description see in XBUCLEO-F103RB usage description above.

RASPBERRY PI

HRDWARE CONNECTION

Usage of Raspberry Pi 3 Model B. Pin connection uses BCM coding of pins.

<table>
<thead>
<tr>
<th>Capacitive Fingerprint Reader</th>
<th>RaspberryPi 3B</th>
</tr>
</thead>
</table>

Date: 2017 Nov. 13
Capacitive Fingerprint Reader

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>VCC</td>
<td>3.3V or 5V</td>
</tr>
<tr>
<td>GND</td>
<td>GND</td>
</tr>
<tr>
<td>TXD</td>
<td>15 (BCM)</td>
</tr>
<tr>
<td>RXD</td>
<td>14 (BCM)</td>
</tr>
<tr>
<td>WAKE</td>
<td>23 (BCM)</td>
</tr>
<tr>
<td>RST</td>
<td>24 (BCM)</td>
</tr>
</tbody>
</table>

Note: before running the device you need to install necessary libraries (WiringPi, bcm2835, python), installation procedure refers to: [Libraries installation for RPi](#)

RASPBERRY PI SERIAL PORT CONFIGURATION

- Since the Raspberry Pi serial port is used for debugging terminal, you need to modify the Raspberry Pi settings. Execute the following:
  
  ```
  sudo raspi-config
  ```

- Select Advanced Options -> Serial -> no to close serial port terminal (there are some differences between different versions of Raspbian, you need check them in internet)

- Open file /boot/config.txt, find the following configuration statements, uncomment it to enable the serial port. If it is missing, add it to the end of the file:
  ```
  enable_uart=1
  ```

- Restart to apply changes.

Note: in the case of Raspberry Pi 3B the serial device number is in ttyS0. In case of Raspberry Pi 2B/zero user need to set the serial device number as ttyAMA0.

PYTHON

Download the code for Raspberry Pi, install all of required libraries and configure serial port.

Input the command "**sudo python3.5 main.py**" or "**sudo ./main.py**" to run the program

(the program is based on python version 3.5 and above).

WIRINGPI

Input the command "**make**", to compile the code and assign name “test” to executable file.
Input the command "sudo ./test" to run the program

EXECUTION RESULTS

The specification of operation methods, command and warning description see in XNUCLEO-F103RB usage description.

You can exit of program by pressing Ctrl+C

Note: If the serial assistant always returns the error message "***ERROR***", then please ensure that the module is powered by 3.3V or 5V, whether the serial wired connection is correct, the module baud rate is 19200 by default, and finally power off, then power on again.
APPENDIX: APPLICATION EXAMPLE

1. **Fingerprint reading application (case A) include devices:** scanner and comparing terminal

2. **Fingerprint reading application (case B) includes devices:** scanner terminal and comparing server
   1) Create database and list for fingerprint
   2) Obtain fingerprints
   3) Upload fingerprint
   4) Comparison
   5) Download results

3. **Online comparison application (case A) includes devices:** ID input terminal, fingerprint database, comparator:
   (1) terminal module
   1) Create database and fill it by fingerprint’s data
2) Input user ID, send ID to server with fingerprints
3) Download from server fingerprint according user ID to terminal, save or keep in buffer
4) Scanning and comparison of fingerprints

4. **Online comparison application (case B) includes devices**: download fingerprint to terminal, compare in terminal.
   (1) terminal module
   1) Create database and fill it by fingerprint’s data
   2) Download the fingerprint by stuff according user permission from server to module
3) Scan user’s fingerprint and compare it with loaded fingerprint

5. **Online comparison application (case C) includes devices:** terminal scanning fingerprint, uploading server for “1:1” comparison
   (1) terminal module
   1) Create database and file it by fingerprint’s data
   2) Scan fingerprint, obtain fingerprint’s characteristics (at fingerprint module)
   3) Upload fingerprint’s characteristics to server
   4) Accept uploaded characteristics and compare with certain characteristics from database
   5) Return comparison result
6. **Online comparison application (case D) includes devices:** terminal scanning fingerprint, uploading server for “1:N” comparison
   (1) terminal module
   1) Create database and fill it by fingerprint’s data
   2) Scan fingerprint, obtain fingerprint’s characteristics (at fingerprint module)
   3) Upload fingerprint’s characteristics to the server.
   4) Accept uploaded characteristics and compare with all characteristics from database
   5) Return comparison’s result.

7. **Fingerprint card application (case A):** loader of fingerprint from database by card ID, fingerprint scanning and comparing terminal
   (1) terminal module
   1) Create database and fill it by fingerprint’s data
   2) Read card’s ID at terminal
   3) Send ID to the server for getting fingerprint characteristics
   4) Download the characteristics to fingerprint module.
5) Compare scanned fingerprint with download one

8. **Fingerprint card application (case B):** fingerprint memory card saver, fingerprint memory card downloader and comparator
   (1) terminal module
   1) Create database and fill it by fingerprint’s data
   2) Save fingerprint characteristics of the user into the memory of card
   3) Use the card to obtain fingerprint characteristics and save them in it.
   4) Compare characteristics of scanned fingerprint and kept in card memory one
5) Delete data from card’s memory after successful comparison.