

Deegou Electronics

Making some electronic modules for quick prototyping

FPC1020

Contents [\[hide\]](#)

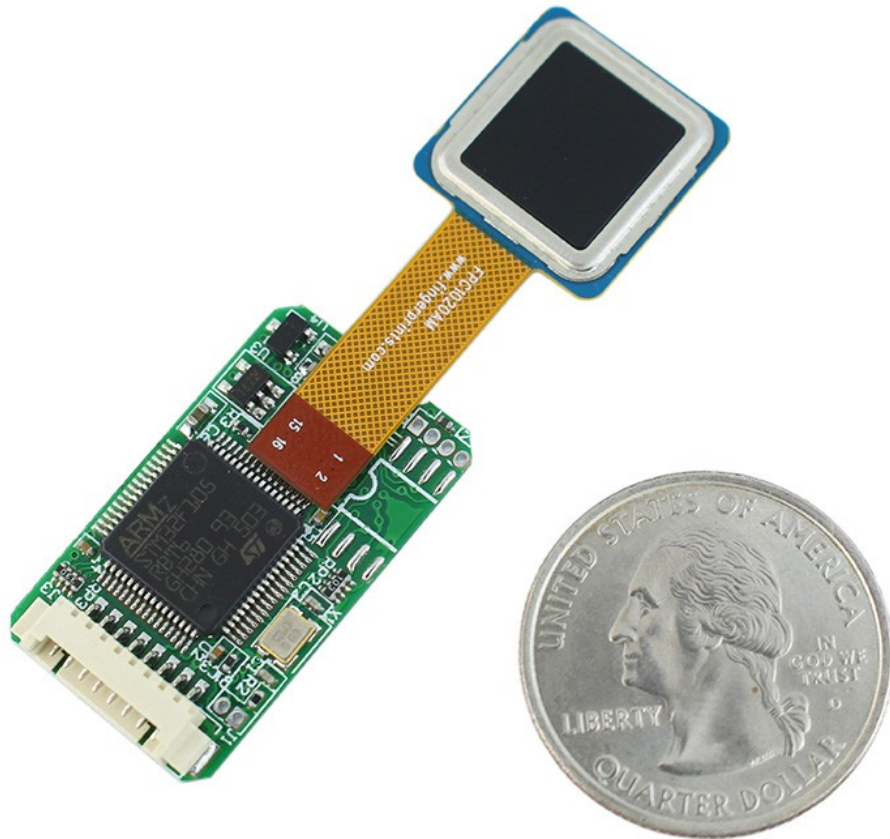
- 1 Description
- 2 Features
- 3 Specifications
- 4 Pin Configuration
- 5 Pin Definitions
- 6 Arduino Demo
 - 6.1 Hooking up
 - 6.2 Installation
 - 6.3 Sketch Code
 - 6.4 Monitor report
- 7 Downloads
- 8 Support

Description

This is a fingerprint sensor kit, it is composed with a fingerprint sensor and a control board.

The fingerprint sensor's mode is FPC1020 from [Fingerprint Cards](#), which has been widely used on smartphones such as Huawei Mate 7. It is based on capacitive sensor technology. It delivers best in class image quality with 256 gray scale values in every single programmable pixel. No matter dry, wet or dirty finger, it can read with its 3D pixel sensing technology. The control board is based on STM32F105, it has a pre-burned program inside for processing the UART commands. Through this board, we can directly drive the fingerprint sensor via serial port.

This fingerprint sensor kit is easy to use. There is an Arduino code provided that allows us to quickly get started. Also, a Grove cable is comes, so that we can quickly add it to the Grove system.



Features

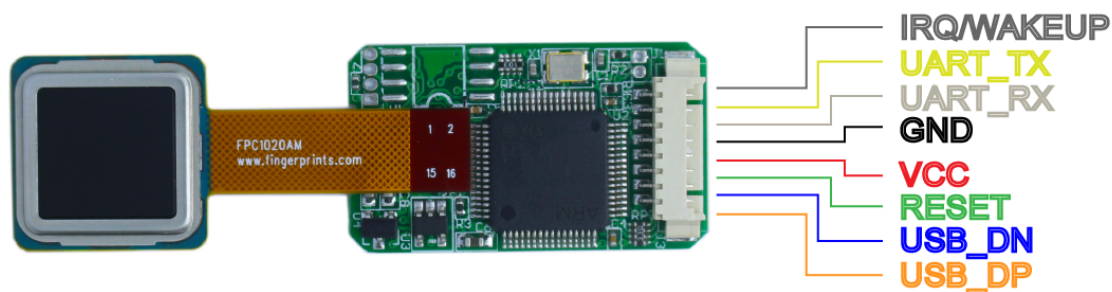
- GROVE Compatible Interface
- FPC1020 CMOS fingerprint sensor
- Best in class imaging quality with 256 true grey scale values in every pixel
- Robust protective coating capable of more than 10 million finger placements
- Full ESD protection to more than $\pm 30\text{kV}$
- 200 byte fingerprint template
- 1:N Identification (One-to-Many)
- 1:1 Verification (One-to-One)
- Auto-learning function (Automatically updating the fingerprint features)
- Security level setting
- TTL serial interface

Specifications

- Resolutions: 508 DPI
- Fingerprint storage capacity: 100 pcs of fingerprint template

- Verification time: < 0.45 sec
- Identification time: < 0.45 sec
- False accept rate (FAR): < 0.0001%
- False reject rate (FRR): < 0.01%
- Interface: Serial UART, 19200 bps (default)
- Baud rate: 9600, 19200, 38400, 57600, 115200 bps
- Working current: < 50mA
- Standby current: < 10 μ A
- Supply voltage: DC 5V
- Digital I/O voltage: 3.3V, 5V
- Operating temperature: - 20 °C ~ 60 °C
- Operating humidity: 20% ~ 80%

Pin Configuration



Pin Definitions

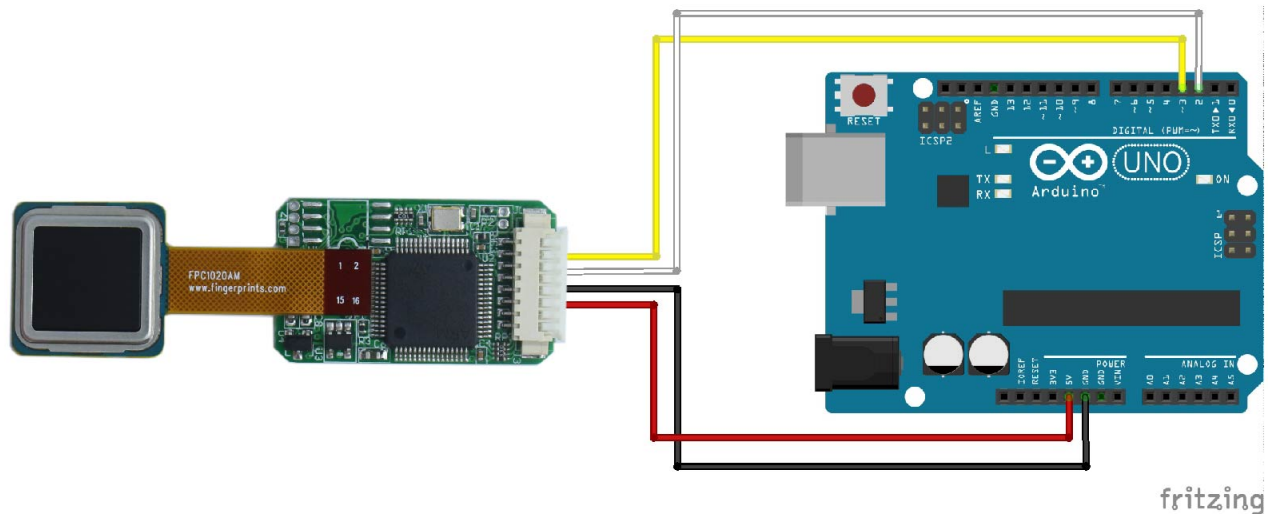
NAME	I/O	DESCRIPTION
IRQ/WAKEUP	OUTPUT	This pin will output a 100us high pulse when a finger is detected
UART_TX	OUTPUT	Transmitter of TTL serial
UART_RX	INPUT	Receiver of TTL serial
GND	POWER	Power GND
VCC	POWER	5V power input
RESET	INPUT	If you want a reset, send a low pulse to this pin

USB_DN	INPUT / OUPUT	Not available for the current version
USB_DP	INPUT / OUTPUT	Not available for the current version

Arduino Demo

Hooking up

There are only four pins that need to be hooked up in order to start using this sensor in a project. One for VCC, one for GND, and two data lines.



- VCC → 5V
- UART_RX → Digital 2
- UART_TX → Digital 3
- GND → GND

Installation

- Download [FPC1020.zip](#)
- Unzip, and add to Arduino/hardware/libraries/

Sketch Code

```
//
```

```
// Demo code for FPC1020 Fingerprint Sensor
// Created by Deray on 2015-10-07.
//

#include <SoftwareSerial.h>
#include <FPC1020.h>

extern unsigned char l_ucFPID;
extern unsigned char rBuf[192]; //Receive return data

#define sw_serial_rx_pin 2 // Connect this pin to TX on the FPC1020
#define sw_serial_tx_pin 3 // Connect this pin to RX on the FPC1020

SoftwareSerial swSerial(sw_serial_rx_pin, sw_serial_tx_pin); // Fingerprint serial (RX, TX
)
FPC1020 Finger(&swSerial);

void setup(){
  Serial.begin(19200);
  pinMode(4, INPUT); // IRQ

  Serial.println("Fingerprint Test ! ");
}

void loop(){
  unsigned int User_ID = 0;
  unsigned char incomingNub;
  unsigned int matchUserID = 0;
  unsigned char rtf = 0;

  while(1){
    Serial.println("===== Menu =====");
    Serial.println("Add a New User ----- 1");
    Serial.println("Fingerprint Matching ----- 2");
    Serial.println("Get User Number and Print All User ID ----- 3 ");
    Serial.println("Delete Assigned User ----- 4");
    Serial.println("Delete All User ----- 5");
    Serial.println("===== End =====");

    unsigned char MODE = 0;

    while(Serial.available()<=0);

    MODE = Serial.read()-0x30;
```

```
switch(MODE){
case 0: // Null
break;

case 1: // Fingerprint Input and Add a New User
MODE = 0;
User_ID = 0;

Serial.println("Please input the new user ID (0 ~ 99).");
while(Serial.available()<=0);
delay(100);
incomingNub = Serial.available();
for(char i=incomingNub; i>=1; i--){
User_ID = User_ID + (Serial.read()-0x30)*pow(10,(i-1));
}

Serial.println("Add Fingerprint, please put your finger on the Fingerprint Sensor.");
rtf = Finger.Enroll(User_ID);

if(rtf == TRUE) {
Serial.print("Success, your User ID is: ");
Serial.println( User_ID , DEC);
}
else if (rtf == FALSE) {
Serial.println("Failed, please try again.");
}
else if( rtf == ACK_USER_OCCUPIED){
Serial.println("Failed, this User ID already exists.");
}
else if( rtf == ACK_USER_EXIST){
Serial.println("Failed, this fingerprint already exists.");
}
delay(2000);
break;

case 2: // Fingerprint Matching
MODE = 0 ;
Serial.println("Match Fingerprint, please put your finger on the Sensor.");

if( Finger.Search()){
Serial.print("Success, your User ID is: ");
Serial.println( l_ucFPID, DEC);
}

else {
```

```
Serial.println("Failed, please try again.");
}
delay(1000);
break;

case 3: // Print all user ID
MODE = 0;
if(Finger.PrintUserID()){
Serial.print("Number of Fingerprint User is:");
unsigned char UserNumb;
UserNumb = (1_ucFPID-2)/3;

Serial.println(UserNumb,DEC);
Serial.println("Print all the User ID:");

for(char i = 0; i < UserNumb; i++){
Serial.println(rBuf[12+i*3],DEC);
}
}

else {
Serial.println("Print User ID Fail!");
}
delay(1000);
break;

case 4: // Delete Assigned User ID
MODE = 0;
User_ID = 0;
Serial.println("Please input the user ID(0 ~ 99) you want to delecte.");
while(Serial.available()<=0);
delay(100);
incomingNub = Serial.available();
for(char i=incomingNub; i>=1; i--){
User_ID = User_ID + (Serial.read()-0x30)*pow(10,(i-1));
}

if(Finger.Delete(User_ID)) {
Serial.println("Delete Fingerprint User Success!");
}
else{
Serial.println("Delete Fingerprint User Fail!");
}
delay(1000);
break;
```

```
case 5: // Delete All User ID

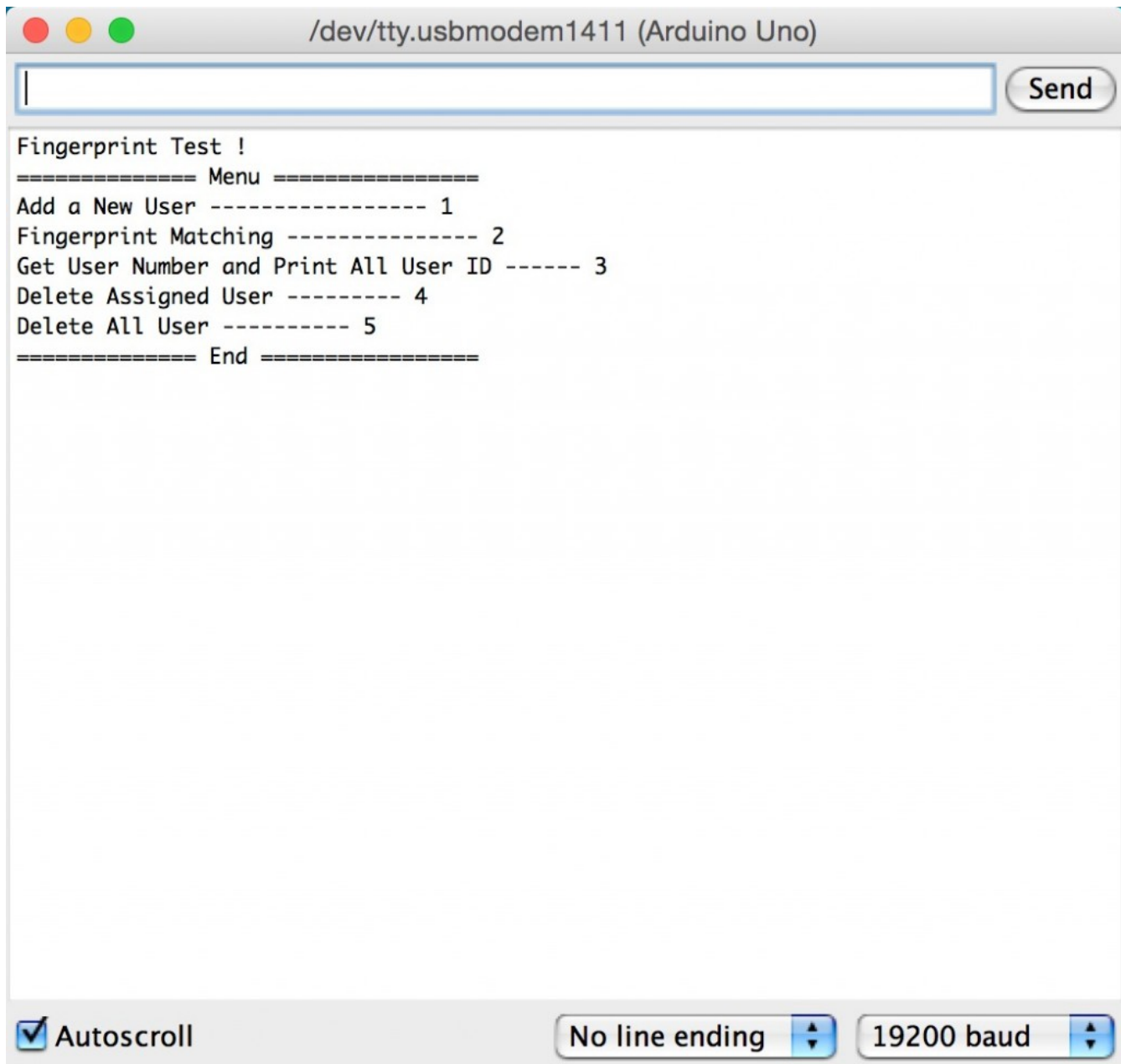
MODE = 0;
unsigned char DeleteFlag = 0;

Serial.println("Delete All Users, Y/N ?");

for(unsigned char i=200; i>0; i--)//wait response info
{
delay(20);
if(Serial.available(>0)
{
DeleteFlag = Serial.read();
break;
}
}

if(DeleteFlag == 'Y' || 'y'){
if(Finger.Clear()){
Serial.println("Delete All Fingerprint User Success!");
}
else{
Serial.println("Delete All Fingerprint User Fail!");
}
}
delay(500);
break;
}
}
```

Monitor report



Downloads

- [Fingerprint_Protocol_All_English.pdf](#)
- [FPC1020_Prodcut_General_Specification.pdf](#)
- [FPC1020_Product_Electric_Specification.pdf](#)
- [8051_MCU_Demo_Code \(Chinese\).zip](#)
- [Arduino_Demo_Code.zip](#)

Support

Any question, document mistakes and suggestions , please contact deray@deegou.com.