FCP900

Module

Product specification

DOC. VERSION 1.0

Startek Engineering Incorporated

April 2017

Content

Revision History

Doc. Version	Revision Description	Date
1.	Initial version	2017/04/21

1. Overview

1.1 General Description

This product is a mini fingerprint algorithm Module and can be used with STARTEK's CMOS optical reader .It has a wide range of applications but focuses mostly on things that need authentication such as computer peripherals, safe boxes, briefcases, car locks, door locks, and can also replace current card systems to further enhance the level of security.

In the new implemented FCP900, it overcomes the problem some other optical readers have in requiring constantly replacing lens coating to speed up fingerprint image scanning. In addition, the image distortion rate of FCP900 is decreased dramatically from 15% of a general CMOS reader to less than 1%. Its Modular Transfer Function (MTF) value has also been improved to more than 40% comparing to 30%~20% for a general CMOS reader.

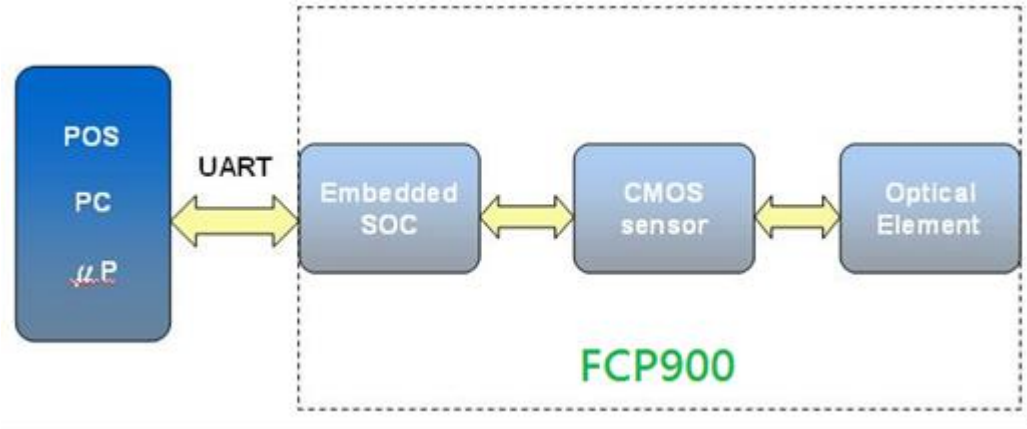
1.2 Features

- Stand Alone: FCP900 can work without connecting to PC.
- High Performance: With high performance CPU, both 1:1 and 1:N matching speed is less than 1 second.
- Small Size: FCP900 is flexible to integrate to any device (P.O.S., Door lock...).
- Cost Effective: FCP900 make your product more competitive.
- Easy Integration: It is very easy to capture fingerprint image, enroll and match fingerprint through RS232/UART interface.
- ISO Standard: ANSI/INCIT 378 Compliant Extractor & Matcher.
- Low Power Consumption: FCP900 can work with battery power supply. Compact and elegant
- Choice to set different security levels for different FRR/FAR demand
- Small template size
- Support verification on various platforms

1.3 Specification

Sensors	CMOS sensor
Scanning Speed	8 Frame/sec
Image Resolution	500 dpi (H), 500 dpi (V)
Image Capture Format	0.5 “ × 0.5 “(H * V)
Gray Level	8 bits/pixel, max 256 gray scales
Image Ratio of Length to Width	1：1
Interface	UART (115200, n, 8, 1) / TTL Level
Power	DC 3.5V (Standby: 50mA, Matching duration: 130mA)
Dimension	63 x 22.4 x 25.4 mm
Weight	25g
Operating Temperature	-5℃ ~ 50℃
Operating Humidity	0~95% Non-condense
Verification speed	1:1, 600 msec, 1:N 900 msec
Fingerprint Template Size	< 1KB
FRR (False Rejection Rate)	1/100
FAR (False Acceptance Rate)	1/100,000
ESD	>15 KV
Template Capacity	1,000

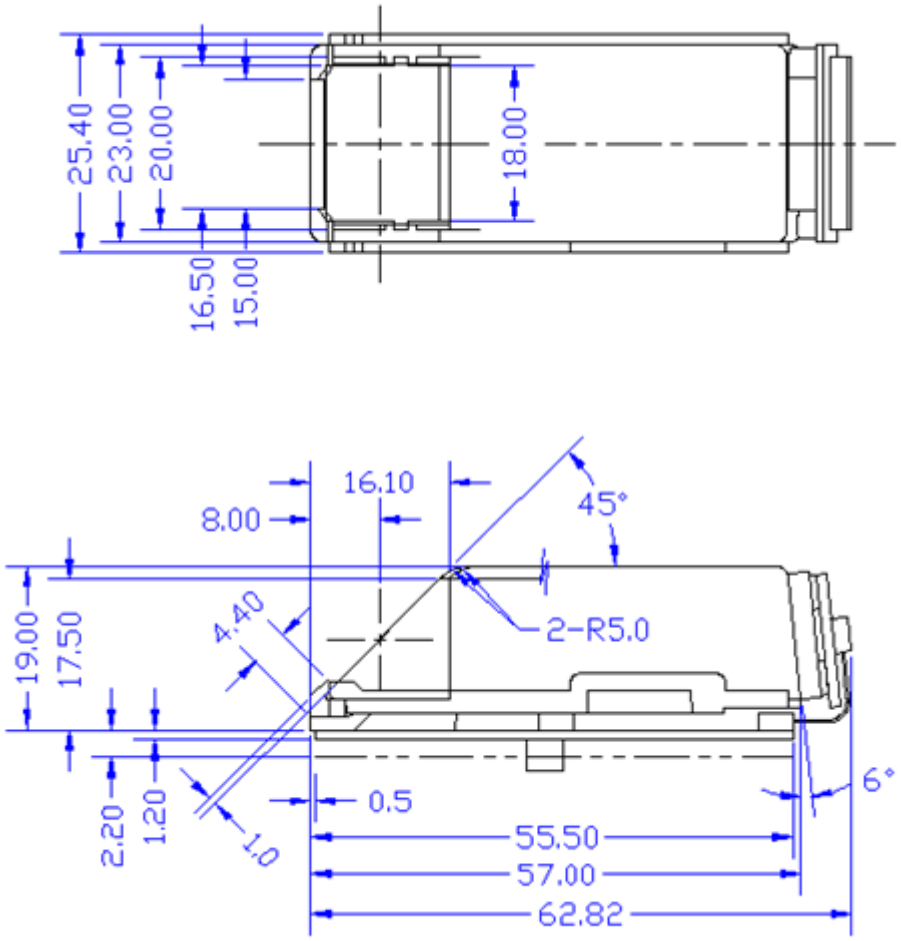
1.4 Block Diagram



1.5 Connetor PIN Description

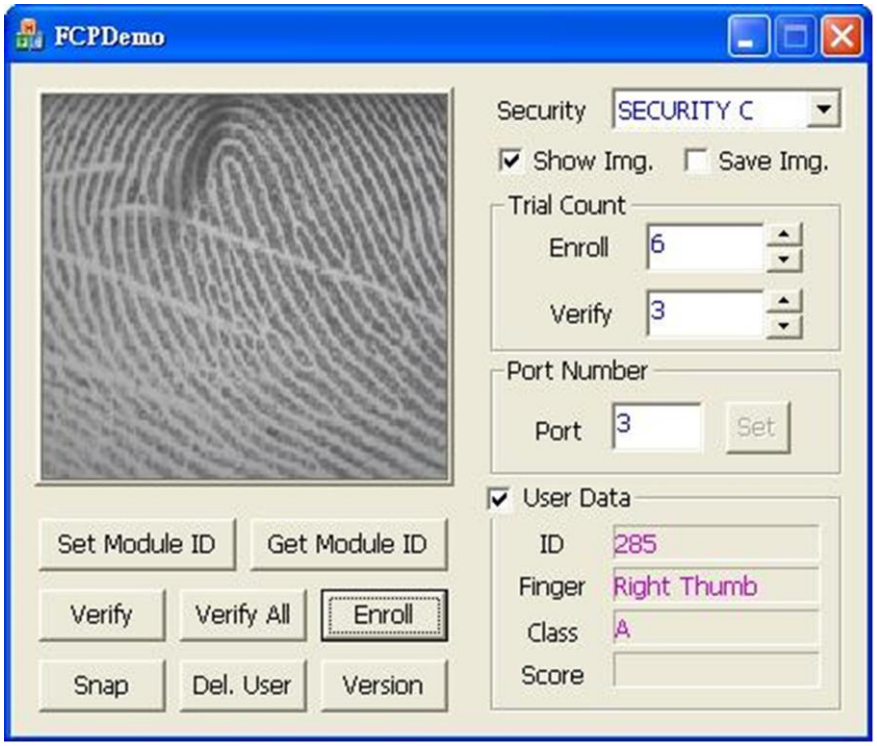
Name	No	States	Description
Null	1	---	---
Null	2	---	---
GND	3	Power	Ground
VCC	4	Power	DC3.5V (Sb:50mA / Max:130mA)
TxDO	5	Singal	3.3V Base TTL UART
RXDO	6	Singal	3.3V Base TTL UART
NRST	7	Singal	Module Reset(Low Active)

1.6 Machanical Dimensions



2. Normal Working Scene of SDK Demo

(Capture fingerprint image for Windows XP)



3. FCP900 communication protocols

Snap: IT9308_Capture

Get Primary Code: IT9308_GetPrimaryCode

IT9308_GetPrimaryCode _Inside

Enroll: IT9308_CreateEnrollHandle

IT9308_Enroll

IT9308_ENROLL_INSIDE

Match: IT9308_CodeMatchEx

IT9308_IDVerify

Image: IT9308_GetImage

Transmitted Data Sequence:

SAP_STX	0	Command	Length	Parameter Data	SAP_ETX	Check sum

TD[ ]: represents the transmitted data sequence

TD[n-1]: the n-th byte of the transmitted data sequence.

TD[0]: the start byte of the transmitted data sequence. Its value is **SAP_STX**.

TD[1]: the separate byte between the start byte and the command byte. The typical value of this byte is 0.

TD[2]: the command byte of the transmitted data sequence. For example, the command **IT9308_CAPTURE** represents to connect the capture device.

TD[3] & **TD[4]:** represents the data length in bytes of the parameter data where. **TD[3]** is the high byte of the data length and **TD[4]** is the low byte of the data length. For example, if the data length is 1000, then **TD[3]=** (BYTE)(1000/256) and **TD[4]=**(BYTE)(1000%256). This length doesn't include the end byte (**SAP_ETX**) and the check sum byte.

TD[5]~TD[L+5-1]: are the parameter data need for this command. For instance, for the command **IT9308_GET_ID**, the parameter data is the user ID.

TD[Total length–2]: the end byte of the transmitted data sequence. Its value is **SAP_ETX**.

TD[Total length–1]: the check sum of the transmitted data sequence. This byte equals the summation of all the bytes from **TD[0]** to **TD[Total length –2]**.

Return Data Sequence:

SAP_STX	0	Resp-Command	Length	Parameter Data	SAP_ETX	Check sum

RD[ ]: represents the return data sequence

RD[n-1]: the n-th byte of the return data sequence.

RD[0]: the start byte of the return data sequence. Its value is **SAP_STX**. **RD[1]:** the separate byte between the start byte and the response-command byte. The typical value of this byte is 0.

RD[2]: the response-command byte of the return data sequence. It represents the responded command of the calling command. For example, when the command **IT9308_CAPTURE** is sent to the device and then the response-command will be **IT9308_RESP_CAPTURE**.

RD[3] & RD[4]: represents the data length in bytes of the return data where **RD[3]** is the high byte of the data length and **RD[4]** is the low byte of the data length. For example, if the data length is 1000, then **RD[3]=** (BYTE)(1000/256) and **RD[4]**=(BYTE)(1000%256). This length doesn't include the end byte (**SAP_ETX**) and the check sum byte.

RD[5]~RD[L+5-1]:are the return data for this command. For instance, for the response-command **IT9308_RESP_CAPTURE**, the return data is **OK** or **FAIL**.

RD[Total length –2]: the end byte of the return data sequence. Its value is **SAP_ETX**.

RD[Total length –1]: the check sum byte of the return data sequence. This byte equals the summation of all the bytes from **RD[0]** to **RD[Total length –2]**.

4. Protocol Example:

IT9308_GetPrimaryCode_Inside

Description:

This command is used to convert the fingerprint image in IT9308 into a 300-byte primary fingerprint code.

Transmitted Data Sequence:

SAP_STX	0	IT9308_GET_PCODE _INSIDE	0 (2 bytes)	SAP_ETX	Check sum

Command: **IT9308_GET_PCODE_INSIDE**

Length: 0 (2 bytes, low bytes is 0x00 and high bytes is 0x00)

Parameter Data: None

Return Data Sequence:

SAP_STX	0	IT9308_RESP_GET _PCODE_INSIDE	2 (2 bytes)	rtn	SAP_ETX	Check sum

Response-Command: **IT9308_RESP_GET_PCODE_INSIDE**

Length: 2 (2 bytes, low byte is 0x02 and high byte is 0x00)

Return Value:

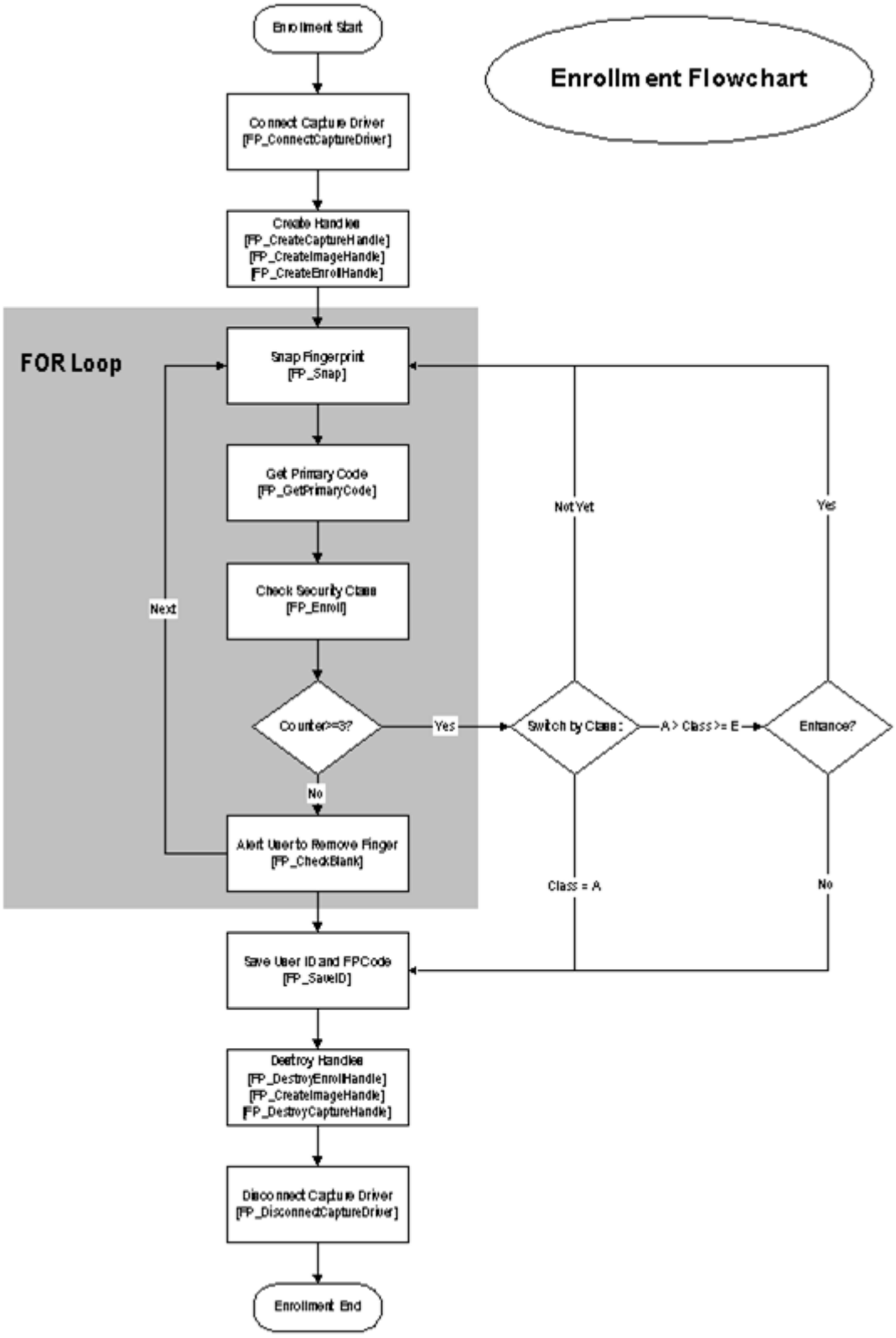
1. rtn: 2-byte return value of this command. (rtn= 256*HiByte+LoByte)
   i. **OK** If succeeds
   ii. **Otherwise** If fails

Remark:

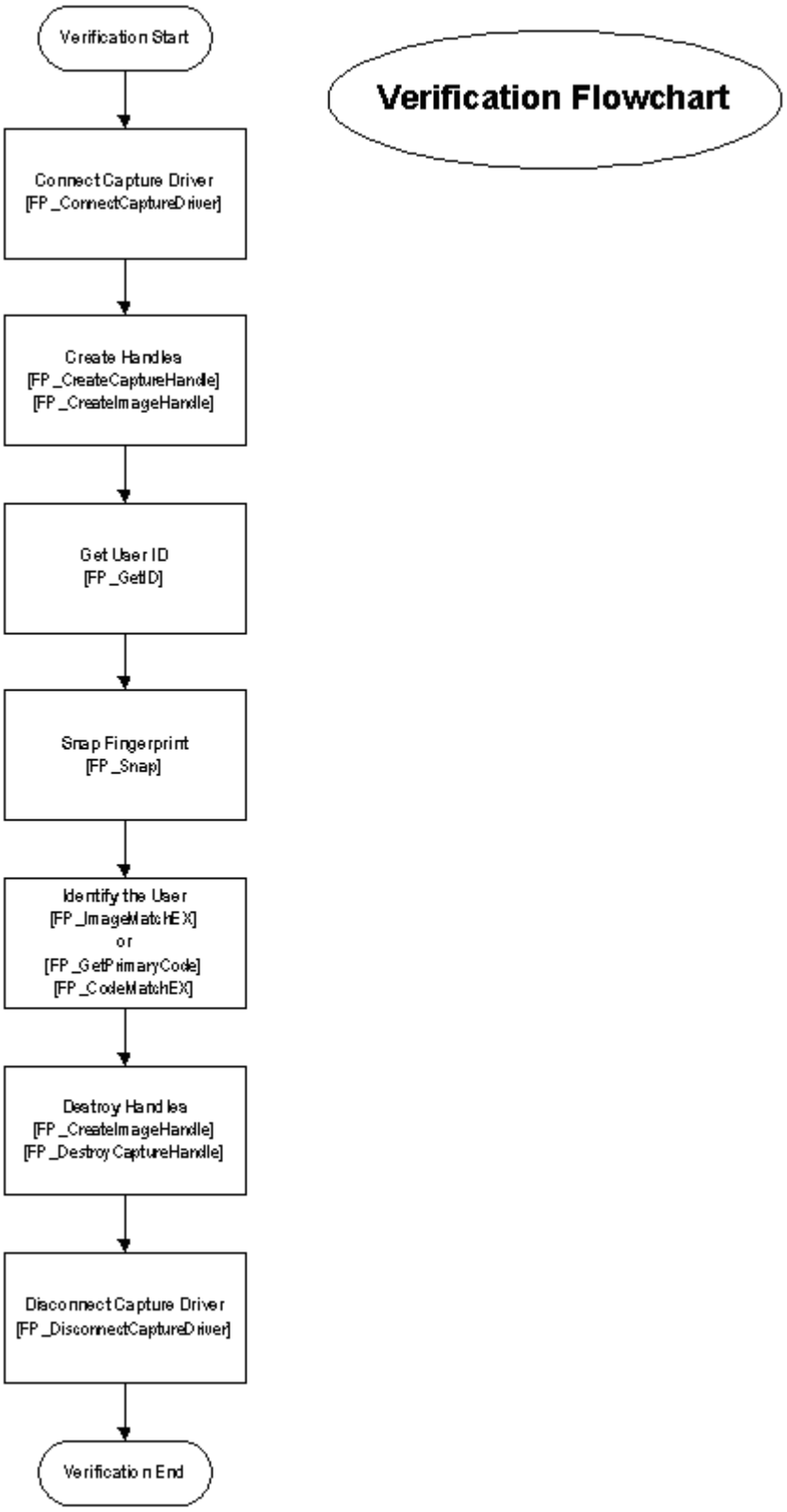
This function is similar to **IT9308_GetPrimaryCode**. But there is no primary code returned. This function only returns whether it succeeds. The converted primary code will be stored in memory temporarily.

5. SDK Process Flowchart

The following diagram shows the simplified enrollment process flow：



The following diagram shows the simplified verification process flow：



6. ANSI/INCITS 378 Approval Authorization Letter

http://fips201ep.cio.gov/apl.php

Approval Authorization Letter

15 September 2008
April Giles
1800 F. Street N.W.
Washington, D.C. 20405

Re: Approval Authorization Letter
Case File Number 723

To: Ed Morris:

The Approval Authority officially approves the following Product/Service. Please add it to the Approved FIPS 201 Products and Services List

Supplier Name:	STARTEK Engineering Inc.
Name of Product or Service:	STARTEK ANSI/INCITS 378 Template Generator for PIV
Part Number:	002B0110
Hardware Version:	N/A
Software Version:	1.0
Firmware Version:	N/A

Restrictions:

- None

Sincerely,

April Giles

April Giles
Approval Authority
202.501.1123
April.giles@gsa.gov