



中印云端指纹识别模块 产品规格书V1 (ZYC-M01)

中印云端



目 录

目录

一、 产品概述.....	3
二、 产品介绍.....	6
三、 应用领域.....	7
四、 硬件接口定义.....	8
五、 模块触控功能说明及应用电路参考设计.....	9
六、 电气参数.....	10
七、 选型规格：	10



一、产品概述

ZYC-M01系列指纹识别模块是一种触摸式指纹识别设备, 由以下两部分组成:

- ZYC-M01指纹识别模块
- 电容式指纹传感器

ZYC-M01系列指纹识别模块具有耐磨、耐腐蚀、耐静电等优势, 模块中集成了中印云端(科技)有限公司科技自主适配PB (PRECISE BIOMATCH™)的指纹识别算法, 该算法在行业内处于领先地位, 长期应用于银行金融业、公安、智能指纹锁等领域, 各项性能指标均位于市场同类产品前列。

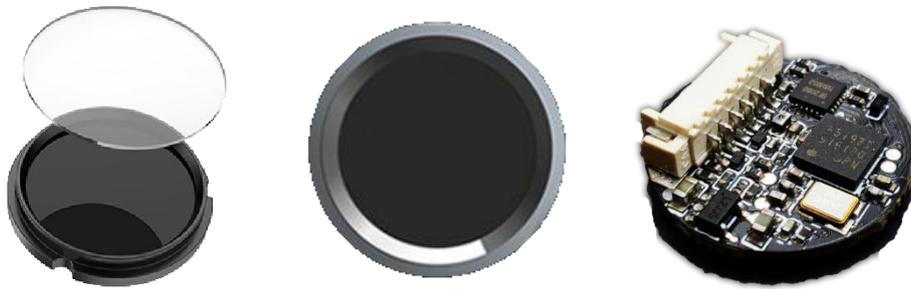
ZYC-M01系列指纹识别模块采用瑞萨高端MCU-RX651系列, RX651的工作效率是M4内核的1.3倍, 标配256KB RAM/1MB Flash, 可扩展至640KB RAM/2MB Flash, 指纹的采集、存储和比对全部在芯片内部处理完成, 处理速度快, 开发接口简单, 便于行业用户的二次开发, 因此可降低产品开发难度, 缩短产品研发周期, 同时通过集成化芯片的方式也实现了指纹模块体积的最小化。

1.1 产品特点

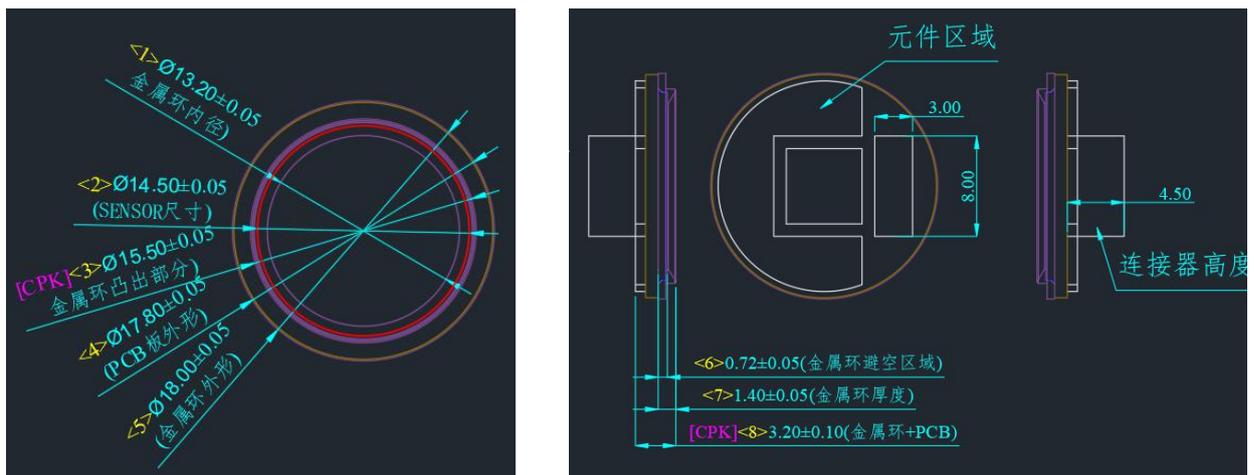
	产品特点及优势
指纹模块	1) 模块采用瑞萨私有安全内核和PB金融级安全算法; 2) 模块化设计、结构简单, 集指纹采集、存储和识别为一体, 完全自行处理, 速度更快、效果更优; 3) 集成度高、体积小、稳定性强、功耗低; 4) 产品稳定性高, 支持零下40℃工作;
指纹识别算法	1) 自主适配PB核心算法, 拥有自主知识产权; 2) 360度旋转识别; 3) 自学习功能;
指纹传感器设计	1) 采用晶圆塑封技术, 具备防雾、防尘、防破坏能力, 有效解决人体静电影响、采像质量高、产品耐用性强; 2) 内置人体感应器件, 具有触摸唤醒功能和低功耗设计, 可有效识别塑胶、硅胶、指模、指套等假手指情况; 3) 指纹模组的晶圆塑封可按用户要求定制颜色;
开发	1) 公开接口代码和指令集, 可实现组装式二次开发, 降低客户开发难度且缩短研发周期; 2) 可为客户提供高效、灵活的二次开发支持, 满足客户所有需求;



1.2 产品外观



ZYC-M01 系列图片



ZYC-M01 立式结构图



1.3 技术指标

指纹图像抓取及比对速度	1:N <400ms (N≤100)
存储容量	标配200 枚指纹数据 (根据用户需要可扩展)
传感器类型	电容触摸式传感器
分辨率	508DPI
FRR.	<1%
FAR.	<0.001%
指纹传感器形状	圆形 (可订做方形)
金属外框结构尺寸	Φ18.0mm (金属环外框直径)
图像灰度等级	8 位灰度
使用寿命	一百万次
电压/电流	3.3V/40mA
静电测试	接触放电8KV/空气放电15KV
数据接口	RS232 (TTL电平)
工作环境	温度: -40° C ~ 85° C 湿度: 40%RH-85%RH (无凝霜)
存储环境	温度: -40° C~105° C 湿度: <85%RH (无凝霜)



二、 产品介绍

ZYC-M01电容指纹识别模块使用电容指纹传感器, 可完成指纹的采集、 比对、 储存以及相关的扩展功能。模块包含硬件和软件 (核心算法及管理程序) 两部分。

2.1 硬件

指纹模块的硬件包含指纹传感器和指纹识别模块。ZYC-M01系列电容指纹识别模块的主芯片采用高性能瑞萨RXV2私有内核120MHz主频, 稳定可靠并能够快速运行指纹识别算法。

2.2 软件 (核心算法及管理程序)

指纹模块的软件包括指纹识别核心算法和管理程序两部分。

2.2.1 算法

中印的指纹算法是基于PB (PRECISE BIOMATCH™) 在纯特征点算法的基础上发展出来的图像和特征点的混合匹配算法。其性能远高于传统的特征点算法。

2.2.2 管理程序

ZYC-M01系列电容指纹识别模块的管理程序, 通过TTL电平的RS232接口与主控单元MCU (或上位机) 按照中印自有通信协议进行交互, 模块接收来自主控单元MCU (或上位机) 的指令, 并执行该指令对应的操作, 操作完成后再将执行结果通过RS232接口返回给主控单元MCU (或上位机); 从而实现指纹处理模块的管理平台。

管理程序的通信接口由若干指令组合而成, 模块的每个功能由主控单元MCU (或上位机) 发送独立的指令来执行, 执行状态通过串口反馈给主控单元MCU进行逻辑交互。通过合理的组合使用接口指令, 可以适用于指纹识别的各种应用场景, 如何实现功能逻辑则完全由主控单元MCU (或上位机) 决定, 方便用户进行二次开发。



三、应用领域

ZYC-M01系列指纹识别模块是目前市面上具备量产能力的,同时也是尺寸最小的指纹模块之一(尺寸可订做),可以为用户提供一体化指纹解决方案。

应用领域	应用产品/方案类型
智能锁	智能指纹门锁、指纹保险箱(柜)、箱包锁
移动终端	指纹手机、平板电脑、笔记本电脑、PDA手持终端和智能可穿戴产品(手表、手环等)
考勤/门禁管理	指纹考勤机、指纹门禁控制器
PC电脑外设	指纹鼠标、指纹键盘、Windows指纹仪
安全存储设备	指纹U盘、指纹硬盘
金融领域	银行指纹柜员管理、银行指纹密码储蓄、指纹密码登录、各类智能信用卡防伪、ATM指纹自动提款、银行指纹保险箱等
计算机及互联网身份认证	计算机及网络安全、会员账号与管理、互联网购物第三方支付等
政府	指纹二代身份证、刑侦、户籍管理、社保、公安部门武器库指纹管理等
工控	工业控制、电梯等
教育	教育白板、智能平板等



四、硬件接口定义

4.1 通讯接口

标准 UART TTL 电平, 支持USB串口;

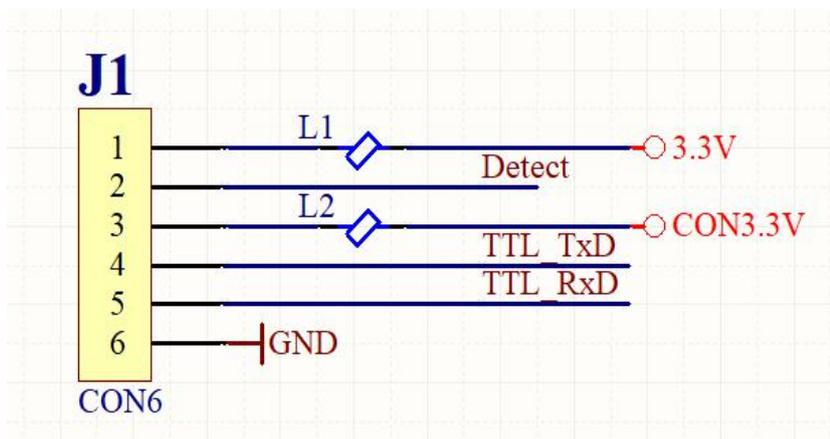
默认波特率115200 bps, 1 起始位, 1 停止位, 3.3V TTL 电平

4.2 连接器类型

1-6P: 6Pin 条形连接器, 间距1.25mm

Micro-USB连接器

4.3 标准管脚说明



PIN 1……6 (照片里从左至右顺序1~6)

- 1) 3.3V: (Touch 触控电路电源)
- 2) Detect 感应上电信号 (手指触摸指纹传感器时输出高电平)
- 3) 3.3V : 用于指纹模块整体供电
- 4) TXD: 指纹模块→主控MCU (或上位机)
- 5) RXD: 主控MCU (或上位机) →指纹模块
- 6) GND

说明:

- 标准品串口为 3.3V 的 TTL 电平。如需与 PC 机连接进行测试, 切换到 USB 接口进行测试;
- **注意:** 1 脚 (3.3V Touch 触控电路电源) 需要一直保持不间断的供电。



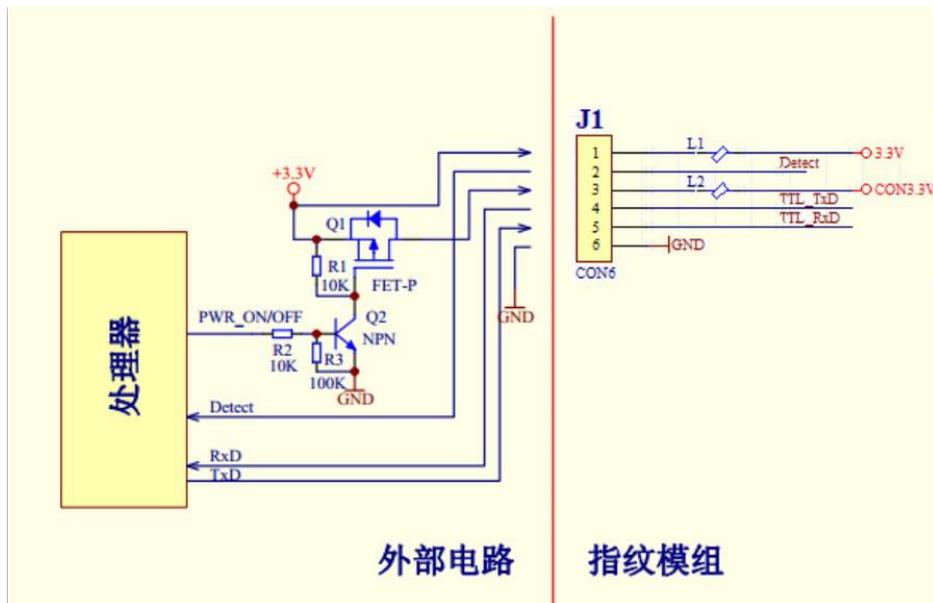
五、模块触控功能说明及应用电路参考设计

5.1 指纹模块触控信号（手指接触感应）原理说明

- 采用指纹芯片检测唤醒功能，功耗低，指纹与触摸唤醒合二为一，判断触发条件更稳定。
- 无手指触摸状态下，Detect信号线为低电平，当手指触摸指纹传感器时该信号触发成高电平，直到手指离开时再变为低电平。
- **使用Detect信号进行二次开发时需要注意以下事项：**
- 在ZYC-M01系列指纹识别模块休眠状态下，作为唤醒信号使用。当手指接触指纹传感器时，触控信号会被激发，继而唤醒系统（给指纹模块上电）。此时二次开发者可无视该信号，并等待指纹模块的正常操作（采集、注册、比对等）完毕后，发送一次复位指令进入模块低功耗状态，完成后并对3.3V 进行掉电，再次触摸指纹模块时，可重复使用该信号。
- 如果锁主控系统是通过触摸键盘唤醒的，则需要在键盘唤醒系统的时候，锁主控端给指纹模块3.3V进行供电，上电150ms后通过串口发送一次复位命令进行指纹头的初始化，然后正常使用。

5.2 模块外部应用电路参考设计（低功耗设计）

降低模块功耗方法通过控制 3.3V 电源是否工作来降低功耗，电路如下图所示：



3.3V 电源控制电路 通用参考原理图不参照型号
(仅供原理参考)

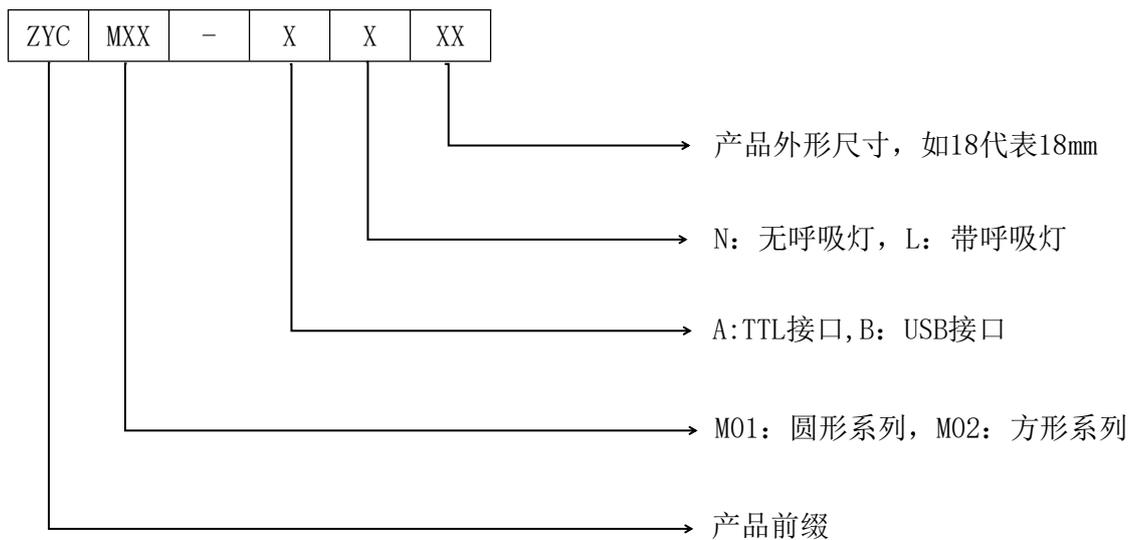
左上部电路作用是通过外部的MCU的PWR_ON/OFF信号控制Q2三极管导通关断时间，从而降低电路功耗。当3.3V电源被切断时，整个模块只有（手指接触）触控信号输出电路在工作，且功耗为8 μ A左右，此Touch触控电源需要保持一直供电状态。



六、电气参数

项目	最小	典型	最大	单位
供电电压	3.0	3.3	3.6	V
掉电待触发电流	-	-	8	uA
工作电流	-	-	40	mA
工作温度	-40	-	85	°C
存储温度	-40	-	105	°C
ESD等级	非接触放电	-	15 K	V
	接触放电	-	8K	V

七、选型规格:



中印云端指纹模组适配的晟元协议1.0指令如下：

指令代码	指令描述	状态
0x01	PS_GetImage	实现
0x29	PS_GetEnrollImage	实现
0x02	PS_GenChar	实现
0x03	PS_Match	实现
0x04	PS_Search	实现
0x05	PS_RegModel	实现
0x06	PS_StoreChar	实现
0x07	PS_LoadChar	实现
0x08	PS_UpChar	实现
0x09	PS_DownChar	实现
0x0a	PS_UpImage	实现
0x0b	PS_DownImage	实现
0x0c	PS_DeletChar	实现
0x0d	PS_Empty	实现
0x0e	PS_WriteReg	实现
0x0f	PS_ReadSysPara	实现
0x31	PS_AutoEnroll	实现
0x32	PS_AutoIdentify	实现
0x12	PS_SetPwd	实现
0x13	PS_VfyPwd	实现
0x14	PS_GetRandomCode	实现
0x15	PS_SetChipAddr	实现
0x16	PS_ReadINFpage	实现
0x18	PS_WriteNotepad	实现
0x19	PS_ReadNotepad	实现
0x1a	PS_BurnCode	实现
0x1b	PS_HighSpeedSearch	实现
0x1d	PS_ValidTempleteNum	实现
0x1f	PS_ReadIndexTable	实现
0x30	PS_Cancle	实现
0x31	PS_AutoEnroll	实现
0x32	PS_AutoIdentify	实现
0x33	PS_Sleep	实现
0x34	PS_GetChipSN	实现
0x35	PS_HandShake	实现
0x36	PS_CheckSensor	实现

指令代码的相关涵义请参考软件开发指南。

1 软件开发指南

1.1 参数表

- 参数表的内容是协议、算法运行的基本参数。整个软件系统都会用到参数表的内容，所以理解并妥善设置参数表对于如何正确使用芯片至关重要；
- 参数表由 DSP 初始化程序在初次上电时设置，并存储于 FLASH 的系统参数存储区，以后每次上电 SOC 初始化程序都要首先将参数表装载到 RAM 中，并根据参数表内容初始化系统寄存器；参数表长度为 64 字（128 字节）；
- 参数表结构如表 1-1 所示：

参数表的初始内容由 ROM 驻留程序或用户程序在系统第一次上电时设置。

表 1-1 系统参数表

类型	序号	中文名称	英文名称	长度(字)	内容与默认值	注释
PART1	1	状态寄存器	SSR	1	0	
	2	传感器类型	SensorType	1	0--15	
	3	指纹库大小	DataBaseSize	1	根据 FLASH 类型自动判别	
PART2	4	安全等级	SecurLevel	1	3	分 5 个等级
	5	设备地址	DeviceAddress	2	0xffffffff	芯片地址，可通过指令设定
	6	数据包大小	CFG_PktSize	1	1	此 8 个寄存器为系统配置表
	7	波特率系数	CFG_BaudRate	1	6	
	8		CFG_VID	1		
	9		CFG_PID	1		
	10	保留		1		
	11	保留		1		
	12	保留		1		
	13	保留		1		
	14	产品型号	ProductSN	4	ASCII 码	设备描述符
	15	软件版本号	SoftwareVersion	4	ASCII 码	
	16	厂家名称	Manufacturer	4	ASCII 码	
	17	传感器名称	SensorName	4	ASCII 码	
	18	密码	PassWord	2	00000000H	默认为 00000000H
	19	Jtag 锁定标志	JtagLockFlag	2	00000000H	
	20	传感器初始化程序入口	SensorInitEntry	1	入口地址	
	21	录入图像程序入口	SensorGetImageEntry	1	入口地址	

类型	序号	中文名称	英文名称	长度(字)	内容与默认值	注释
	22	保留		27		
PART3	23	参数表有效标志	ParaTableFlag	1	0x1234	

- 参数表位于系统参数存储区第 1 页；
- 参数表在芯片上电时从 flash 装载到 RAM 中，结构与顺序不作任何改变；
- 参数表详解：

1) 状态寄存器 SSR

Reset Value: 0x0000

长度: 1 word

属性: 只读

用途: 系统状态指示

读取指令: PS_ReadSysPara, 详见指令说明

格式: 详见表 1-2

表 1-2 状态寄存器格式

第 15~4 位	第 3 位	第 2 位	第 1 位	第 0 位
Reserved	ImgBufStat	PWD	Pass	Busy

注：

- Busy: 占 1 位，置“1”表示系统正在执行命令，“0”表示系统空闲；
- Pass: 占 1 位，置“1”表示指纹验证通过；
- PWD: 占一位，置“1”表示设备握手口令通过验证；
- ImgBufStat: 占一位，置“1”表示指纹图像缓冲区存在有效指纹图像。

2) 传感器类型 SensorType

Reset Value: 0x0000

长度: 1 word

属性: 只读

用途: 表示传感器驱动类型

读取指令: PS_ReadSysPara, 详见指令说明

3) 指纹库大小 DataBaseSize

Reset Value: According to FLASH

长度: 1 word

属性: 只读

用途: 指纹库容量指示

读取指令: PS_ReadSysPara, 详见指令说明

4) 安全等级 SecurLevel

8) USB ID	CFG_VID
Reset Value:	0x0453
长度:	1 word
属性:	只读
用途:	USB 内嵌协议 VID
读取指令:	PS_ReadINFpage, 详见指令说明
9) USB ID	CFG_PID
Reset Value:	0x9005
长度:	1 word
属性:	只读
用途:	USB 内嵌协议 PID
读取指令:	PS_ReadINFpage, 详见指令说明
10) 产品型号	ProductSN
Reset Value:	第一次上电初始化值
长度:	4 words
属性:	只读
用途:	指示产品型号
读取指令:	PS_ReadINFpage, 详见指令说明
11) 软件版本号	SoftwareVersion
Reset Value:	第一次上电初始化值
长度:	4 words
属性:	只读
用途:	指示软件版本号
读取指令:	PS_ReadINFpage, 详见指令说明
12) 厂家名称	Manufacturer
Reset Value:	第一次上电初始化值
长度:	4 words
属性:	只读
用途:	指示厂家名称
读取指令:	PS_ReadINFpage, 详见指令说明
13) 传感器名称	SensorName
Reset Value:	第一次上电初始化值
长度:	4 words

属性:	只读
用途:	指示传感器名称
读取指令:	PS_ReadINFpage, 详见指令说明
14) 密码	PassWord
Reset Value:	0x00000000
长度:	2 words
属性:	读/写
用途:	握手口令, 口令通过系统才能响应
读取指令:	PS_ReadINFpage, 详见指令说明
设置指令:	PS_SetPwd, 详见指令说明
15) JTAG 锁止标志	JtagLockFlag
Reset Value:	0x00000000
长度:	2 words
属性:	只读
用途:	第一次上电时写入特定的值将关闭 JTAG 端口
读取指令:	PS_ReadINFpage, 详见指令说明
16) 传感器初始化入口	SensorInitEntry
Reset Value:	保留
长度:	1 word
属性:	只读
用途:	系统根据该值调用传感器初始化程序, 保留
读取指令:	PS_ReadINFpage, 详见指令说明
17) 传感器图像获取入口	SensorGetImageEntry
Reset Value:	保留
长度:	1 word
属性:	只读
用途:	系统根据该值调用传感器采集图像程序, 保留
读取指令:	PS_ReadINFpage, 详见指令说明
18) 参数表有效标志	ParaTableFlag
Reset Value:	0x1234
长度:	1 word
属性:	只读
用途:	若该域的值是 0x1234, 则表示参数表已经初始化; 若该域的值是

0x0204，则表示系统只对参数表的 PART1 部分进行初始化；该域若为其他值，系统将初始化参数表。

读取指令： PS_ReadINFpage，详见指令说明

1.2 系统参数存储区结构

- 系统参数存储区结构如表 1-3 所示：

表 1-3 系统参数存储区结构

页号	内容	注解
0	保留	
1	参数表	
2	用户记事本	
3	保留	
4	保留	
5	保留	
6	保留	
7	指纹库索引表	可供索引 1024 枚指纹

- 系统参数存储区分为 8 页，每页 512 字节。

1.3 用户记事本

在 FLASH 中开辟了一个 512 字节的存储区域作为用户记事本，该记事本逻辑上被分成 16 页，每页 32 字节。上位机可以通过 PS_WriteNotepad 指令和 PS_ReadNotepad 指令访问任意一页。注意写记事本某一页的时候，该页 32 字节的内容被整体写入，原来的内容被覆盖。

1.4 缓冲区与指纹库

芯片内设有一个 72K 字节的图像缓冲区和两个特征文件缓冲区。其中，旧版算法特征文件缓冲区是 512 bytes（每个指纹特征文件大小为 256 bytes），新版算法特征文件缓冲区是 768 bytes（每个指纹特征文件大小为 384 bytes）。本文都是以 384 bytes 的指纹特征文件为例，进行叙述。特征文件缓冲区既可以用于存放普通特征文件也可以用于存放模板特征文件。通过 UART 口上传或下载图像时为了加快速度，只用到像素字节的高四位，即将两个像素合成一个字节传送。通过 USB 口则是整 8 位像素。

指纹库容量根据挂接的 FLASH 容量不同而改变，系统会自动判别。指纹模板按照序号存放，序号定义为：0~N-1（N 指指纹库容量）。用户只能根据序号访问指纹库内容。

1.5 特征与模板

指纹特征文件大小为 384 字节，包含特征点信息与总体信息；模板大小为 768 字节，是两个相同指纹

特征之和。

特征文件结构：

- 单个特征文件的特征点个数上限为 50 个，特征文件占 384 字节，其中头 56 字节为文件头，包含总体信息；后 328 字节存放 82 个特征点的信息，每个特征点占 4 字节。
- 文件头格式如表 4-4 所示：

表 1-4 特征文件头格式

第 0~5 字节	第 6~39 字节	第 40~43 字节	第 44~55 字节
标志、类型、特征质量、特征个数、序号	背景表 34 字节	2 个中心点坐标	系统保留

注：

- 1) 标志：1 字节。特征文件标志，用于区分不同传感器或不同算法生成的特征文件；存到数据库时标志域不能为 0，若为 0 则表示该特征文件无效或已被删除；
 - 2) 类型：1 字节。指特征文件类型。0 表示该特征文件仅含文件头，1 表示精简特征，2 表示完整特征；
 - 3) 特征质量：1 字节。指特征的可靠度，从 0~100，分数越大质量越高；
 - 4) 个数：1 字节。从 5~82，指特征点个数。最少 5 个，最多 82 个；
 - 5) 序号：2 字节。搜索辅助用途；
 - 6) 背景表：34 字节。背景压缩表；
 - 7) 奇异点坐标：4 字节。包含两个中心点的 x,y 坐标；
 - 8) 系统保留字节：12 字节。
- 特征单元结构

每个特征单元占 4 个字节（32bits），格式如表 1-5 所示：

表 1-5 特征单元格式

第 31~23 位	第 22~14 位	第 13~5 位	第 4~1 位	第 0 位
x	y	角度	特征点质量	属性

1.6 ROM 及传感器驱动

ROM 内嵌了完整的指纹识别算法。传感器驱动为 synochip 提供，用户可自行开发相关应用层程序。

1.7 口令与地址

指纹模块系统默认口令为 0，若默认口令未被修改，则 USB 通讯时系统不要求验证口令，上位机可以直接与芯片通讯；若通过 URAT 通讯或口令被修改，则上位机与芯片通讯的第一个指令必须是验证口令，只有口令验证通过后，芯片才接收其他指令。

芯片的默认地址为 0xffffffff，可通过指令修改，数据包地址域必须与该地址相配，命令包/数据包才被系统接收。

1.8 上电握手信号

指纹模块系统在上电后，会通过串口发送一个 0x55 信号。主机在等待指纹模块系统初始化时，可以通过接受握手信号，提前进入工作状态。

2 指令集

1) PS_GetImage

- 指令代码: 01H
- 功能: 验证用获取图像

2) PS_GetEnrollImage

- 指令代码: 29H
- 功能: 注册用获取图像

3) PS_GenChar

- 指令代码: 02H
- 功能: 根据原始图像生成指纹特征存于特征文件缓冲区

4) PS_Match

- 指令代码: 03H
- 功能: 精确比对特征文件缓冲区中的特征文件

5) PS_Search

- 指令代码: 04H
- 功能: 以特征文件缓冲区中的特征文件搜索整个或部分指纹库

6) PS_RegModel

- 指令代码: 05H
- 功能: 将特征文件合并生成模板存于特征文件缓冲区

7) PS_StoreChar

- 指令代码: 06H
- 功能: 将特征缓冲区中的文件储存在 flash 指纹库中

8) PS_LoadChar

- 指令代码: 07H
- 功能: 从 flash 指纹库中读取一个模板到特征缓冲区

9) PS_UpChar

- 指令代码: 08H
- 功能: 将特征缓冲区中的文件上传给上位机

10) PS_DownChar

- 指令代码: 09H
-

-
- 功能：从上位机下载一个特征文件到特征缓冲区

11) PS_UpImage

- 指令代码：0aH
- 功能：上传原始图像

12) PS_DownImage

- 指令代码：0bH
- 功能：下载原始图像

13) PS_DeletChar

- 指令代码：0cH
- 功能：删除 flash 指纹库中的一个特征文件

14) PS_Empty

- 指令代码：0dH
- 功能：清空 flash 指纹库

15) PS_WriteReg

- 指令代码：0eH
- 功能：写 SOC 系统寄存器

16) PS_ReadSysPara

- 指令代码：0fH
- 功能：读系统基本参数

17) PS_AutoEnroll

- 指令代码：31H
- 功能：自动注册模板

18) PS_AutoIdentify

- 指令代码：32H
- 功能：自动验证指纹

19) PS_SetPwd

- 指令代码：12H
- 功能：设置设备握手口令

20) PS_VfyPwd

- 指令代码：13H
- 功能：验证设备握手口令

21) PS_GetRandomCode

- 指令代码：14H
-

-
- 功能：采样随机数
- 22) PS_SetChipAddr**
- 指令代码：15H
 - 功能：设置芯片地址
- 23) PS_ReadINFpage**
- 指令代码：16H
 - 功能：读取 FLASH Information Page 内容
- 24) PS_Port_Control**
- 指令代码：17H
 - 功能：通讯端口（UART/USB）开关控制
- 25) PS_WriteNotepad**
- 指令代码：18H
 - 功能：写记事本
- 26) PS_ReadNotepad**
- 指令代码：19H
 - 功能：读记事本
- 27) PS_BurnCode（该指令为烧写片外 FLASH 代码）**
- 指令代码：1aH
 - 功能：烧写片内 FLASH
- 28) PS_HighSpeedSearch**
- 指令代码：1bH
 - 功能：高速搜索 FLASH
- 29) PS_GenBinImage**
- 指令代码：1cH
 - 功能：生成二值化指纹图像
- 30) PS_ValidTempleteNum**
- 指令代码：1dH
 - 功能：读有效模板个数
- 31) PS_UserGPIOCommand**
- 指令代码：1eH
 - 功能：用户 GPIO 控制命令
- 32) PS_ReadIndexTable**
- 指令代码：1fH
-

-
- 功能：读索引表

33) PS_Cancle

- 指令代码：30H
- 功能：取消指令

34) PS_AutoEnroll

- 指令代码：31H
- 功能：自动注册模块指令
-

35) PS_AutoIdentify

- 指令代码：32H
- 功能：自动验证指纹指令
-

36) PS_Sleep

- 指令代码：33H
- 功能：休眠指令
-

37) PS_GetChipSN

- 指令代码：34H
- 功能：获取芯片唯一序列号
-

38) PS_HandShake

- 指令代码：35H
- 功能：握手指令
-

39) PS_CheckSensor

- 指令代码：36H
 - 功能：校验传感器
 -
-

3 指令格式详解

指纹模块挂接必要的外围电路后即可构成完整的指纹识别模块，模块始终处于从属地位（Slave mode），主机（Host）需要通过不同的指令让模块完成各种功能。主机的指令、模块的应答以及数据交换都是按照规定格式的数据包来进行的。主机必须按照下述格式封装要发送的指令或数据，也必须按下述格式解析收到的数据包。

3.1 指令包/数据包格式

指令/数据包共分为三类：

包标识=01：命令包。

包标识=02：数据包，且有后续包。

包标识=08：最后一个数据包，即结束包。

所有的数据包都要加包头：0xEF01。

- 01 命令包格式：

表 3-1 命令包格式

名称	包头	芯片地址	包标识	包长度	指令	参数 1	...	参数 N	校验和
字节数	2 bytes	4 bytes	1 byte	2 bytes	1 byte				2 bytes
内容	0xEF01	xxxx	01	N=					

- 02 数据包格式：

表 3-2 数据包格式

名称	包头	芯片地址	包标识	包长度	数据	校验和
字节数	2bytes	4bytes	1 byte	2 bytes	N bytes	2 bytes
内容	0xEF01	xxxx	02			

- 08 结束包格式：

表 3-3 结束包格式

名称	包头	芯片地址	包标识	包长度	数据	校验和
字节数	2bytes	4bytes	1 byte	2 bytes	N bytes	2 bytes
内容	0xEF01	xxxx	08			

- ◆ 数据包不能单独进入执行流程，必须跟在指令包或应答包后面。
- ◆ 下传或上传的数据包格式相同。
- ◆ 包长度 = 包长度至校验和（指令、参数或数据）的总字节数，包含校验和，但不包含包长度本身的字节数。

- ◆ 校验和是从包标识至校验和之间所有字节之和，超出 2 字节的进位忽略。
- ◆ 芯片地址在没有生成之前为缺省的 0xffffffff，一旦上位机通过指令生成了芯片地址，则所有的数据包都必须按照生成的地址收发。芯片将拒绝地址错误的数据包。
- ◆ 对于多字节的高字节在前低字节在后（如 2bytes 的 00 06 表示 0006，而不是 0600）。

3.2 指令应答

应答是将有关命令执行情况与结果上报给上位机，应答包含有参数，并可跟后续数据包。上位机只有在收到 SOC 的应答包后才能确认 SOC 收包情况与指令执行情况。

- 应答包格式：

表 3-4 应答包格式

名称	包头	芯片地址	包标识	包长度	确认码	返回参数	校验和
字节数	2bytes	4bytes	1 byte	2 bytes	1 byte	N bytes	2 bytes
内容	0xEF01		07				

- ◆ 确认码定义：

00H: 表示指令执行完毕或 OK;

01H: 表示数据包接收错误;

02H: 表示传感器上没有手指;

03H: 表示录入指纹图像失败;

04H: 表示指纹图像太干、太淡而生不成特征;

05H: 表示指纹图像太湿、太糊而生不成特征;

06H: 表示指纹图像太乱而生不成特征;

07H: 表示指纹图像正常，但特征点太少（或面积太小）而生不成特征;

08H: 表示指纹不匹配;

09H: 表示没搜索到指纹;

0aH: 表示特征合并失败;

0bH: 表示访问指纹库时地址序号超出指纹库范围;

0cH: 表示从指纹库读模板出错或无效;

0dH: 表示上传特征失败;

0eH: 表示模块不能接收后续数据包;

0fH: 表示上传图像失败;

10H: 表示删除模板失败;

11H: 表示清空指纹库失败;

12H: 表示不能进入低功耗状态;

13H: 表示口令不正确;

14H: 表示系统复位失败;

15H: 表示缓冲区内没有有效原始图而生不成图像;

16H: 表示在线升级失败;

17H: 表示残留指纹或两次采集之间手指没有移动过;

18H: 表示读写 FLASH 出错;

f0H: 有后续数据包的指令, 正确接收后用 0xf0 应答;

f1H: 有后续数据包的指令, 命令包用 0xf1 应答;

f2H: 表示烧写内部 FLASH 时, 校验和错误;

f3H: 表示烧写内部 FLASH 时, 包标识错误;

f4H: 表示烧写内部 FLASH 时, 包长度错误;

f5H: 表示烧写内部 FLASH 时, 代码长度太长;

f6H: 表示烧写内部 FLASH 时, 烧写 FLASH 失败;

19H: 随机数生成失败;

1aH: 无效寄存器号;

1bH: 寄存器设定内容错误号;

1cH: 记事本页码指定错误;

1dH: 端口操作失败;

1eH: 自动注册 (enroll) 失败;

1fH: 指纹库满;

20H: 设备地址错误;

21H: 密码有误;

22 H: 指纹模板非空;

23 H: 指纹模板为空;

24 H: 指纹库为空;

25 H: 录入次数设置错误;

26 H: 超时;

27 H: 指纹已存在;

28 H: 指纹模板有关联;

29 H: 传感器初始化失败;

2AH—efH: Reserved。

指令只能由上位机下给模块, 模块向上位机应答。

系统上电复位后将首先检查默认的设备握手口令是否被修改，若未被修改，则系统认为上位机没有验证口令的需求，SOC 直接进入正常工作状态；若已被修改，则必须首先验证设备握手口令，口令通过后 SOC 才进入正常工作状态。

3.3 指令详解

3.3.1 验证用获取图像 PS_GetImage

- 功能说明： 验证指纹时，探测手指，探测到后录入指纹图像存于图像缓冲区。返回确认码表示：录入成功、无手指等。
- 输入参数： none
- 返回参数： 确认字
- 指令代码： 01H
- 指令包格式：

表 3-5 录入图像指令包格式

包头	芯片地址	包标识	包长度	指令码	校验和
2bytes	4bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	01H	0003H	01H	0005H

- 应答包格式：

表 3-6 录入图像指令应答包格式

包头	芯片地址	包标识	包长度	确认码	校验和
2bytes	4bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	0003H	xxH	sum

注：确认码=00H 表示获取图像成功；
确认码=01H 表示收包有错；
确认码=02H 表示传感器上无手指；
确认码=03H 表示获取图像不成功；
sum 指校验和。

3.3.2 注册用获取图像 PS_GetEnrollImage

- 功能说明： 注册指纹时，探测手指，探测到后录入指纹图像存于图像缓冲区。返回确认码表示：录入成功、无手指等。
- 输入参数： none
- 返回参数： 确认字
- 指令代码： 29H

- 指令包格式:

表 3-7 录入图像指令包格式

包头	芯片地址	包标识	包长度	指令码	校验和
2bytes	4bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	01H	0003H	29H	002DH

- 应答包格式:

表 3-8 录入图像指令应答包格式

包头	芯片地址	包标识	包长度	确认码	校验和
2bytes	4bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	0003H	xxH	sum

注: 确认码=00H 表示获取图像成功;

确认码=01H 表示收包有错;

确认码=02H 表示传感器上无手指;

确认码=03H 表示获取图像不成功;

sum 指校验和。

3.3.3 生成特征 PS_GenChar

- 功能说明: 将图像缓冲区中的原始图像生成指纹特征文件存于特征文件缓冲区。
- 输入参数: BufferID (正整数)
- 返回参数: 确认字
- 指令代码: 02H
- 指令包格式:

表 3-9 生成特征指令包格式

包头	芯片地址	包标识	包长度	指令码	缓冲区号	校验和
2 bytes	4bytes	1 byte	2 bytes	1 byte	1 byte	2 bytes
0xEF01	xxxx	01H	0004H	02H	BufferID	sum

注: 在注册过程中, BufferID 表示按第几次手指; 其他情况中, BufferID 有相应的默认值。

- 应答包格式:

表 3-10 生成特征指令应答包格式

包头	芯片地址	包标识	包长度	确认码	校验和
2bytes	4bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	0003H	xxH	sum

注: 确认码=00H 表示生成特征成功;

确认码=01H 表示收包有错;

确认码=06H 表示指纹图像太乱而生不成特征;

确认码=07H 表示指纹图像正常, 但特征点太少而生不成特征;

确认码=15H 表示图像缓冲区内没有有效原始图而生不成图像；

确认码=28H 表示当前指纹模板与之前模板之间有关联；

sum 指校验和。

3.3.4 精确比对两枚指纹特征 PS_Match

- 功能说明： 精确比对特征文件缓冲区中的特征文件。
- 输入参数： none
- 返回参数： 确认字， 比对得分
- 指令代码： 03H
- 指令包格式：

表 3-11 精确比对两枚指纹特征指令包格式

包头	芯片地址	包标识	包长度	指令码	校验和
2bytes	4bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	01H	0003H	03H	0007H

- 应答包格式：

表 3-12 精确比对两枚指纹特征指令应答包格式

包头	芯片地址	包标识	包长度	确认码	得分	校验和
2bytes	4bytes	1 byte	2 bytes	1 byte	2 bytes	2 bytes
0xEF01	xxxx	07H	0005H	xxH	xxH	sum

注： 确认码=00H 表示指纹匹配；

确认码=01H 表示收包有错；

确认码=08H 表示指纹不匹配；

sum 指校验和。

3.3.5 搜索指纹 PS_Search

- 功能说明： 以特征文件缓冲区中的特征文件搜索整个或部分指纹库。若搜索到，则返回页码。
- 输入参数： BufferID（默认为 1）， StartPage（起始页）， PageNum（页数）
- 返回参数： 确认字， 页码（相配指纹模板）
- 指令代码： 04H
- 指令包格式：

表 3-13 搜索指纹指令包格式

包头	芯片地址	包标识	包长度	指令码	缓冲区号	参数	参数	校验和
2 bytes	4bytes	1 byte	2 bytes	1 byte	1 byte	2 bytes	2 bytes	2 bytes

包头	芯片地址	包标识	包长度	指令码	缓冲区号	参数	参数	校验和
0xEF01	xxxx	01H	0008H	04H	BufferID	StartPage	PageNum	sum

注：BufferID 默认为 1，以特征文件缓冲区中指纹模板搜索整个或部分指纹库。

- 应答包格式：

表 3-14 搜索指纹指令应答包格式

包头	芯片地址	包标识	包长度	确认码	页码	得分	校验和
2 bytes	4bytes	1 byte	2 bytes	1 byte	2 bytes	2 bytes	2 bytes
0xEF01	xxxx	07H	07H	xxH	PageID	MatchScore	sum

注：确认码=00H 表示搜索到；

确认码=01H 表示收包有错；

确认码=09H 表示没搜索到；此时页码与得分为 0；

确认码=17H 表示残留指纹或两次采集之间手指没有移动过；

sum 指校验和。

3.3.6 合并特征（生成模板）PS_RegModel

- 功能说明： 将特征文件融合后生成新模板，结果存于特征文件缓冲区中。
- 输入参数： none
- 返回参数： 确认字
- 指令代码： 05H
- 指令包格式：

表 3-15 合并特征（生成模板）指令包格式

包头	芯片地址	包标识	包长度	指令码	校验和
2 bytes	4bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	01H	0003H	05H	0009H

- 应答包格式：

表 3-16 合并特征（生成模板）指令应答包格式

包头	芯片地址	包标识	包长度	确认码	校验和
2 bytes	4bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	0003H	xxH	sum

注：确认码=00H 表示合并成功；

确认码=01H 表示收包有错；

确认码=0aH 表示合并失败（两枚指纹不属于同一手指）；

sum 指校验和。

3.3.7 储存模板 PS_StoreChar

- 功能说明： 将特征文件缓冲区中的模板文件存到 PageID 号 flash 数据库位置。
- 输入参数： BufferID（默认为 1），PageID（指纹库位置号）
- 返回参数： 确认字
- 指令代码： 06H
- 指令包格式：

表 3-17 储存模板指令包格式

包头	芯片地址	包标识	包长度	指令码	缓冲区号	位置号	校验和
2 bytes	4bytes	1 byte	2 bytes	1 byte	1 byte	2 bytes	2 bytes
0xEF01	xxxx	01H	0006H	06H	BufferID	PageID	sum

注： BufferID 默认为 1。

- 应答包格式：

表 3-18 储存模板指令应答包格式

包头	芯片地址	包标识	包长度	确认码	校验和
2 bytes	4bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	0003H	xxH	sum

注： 确认码=00H 表示储存成功；

确认码=01H 表示收包有错；

确认码=0bH 表示 PageID 超出指纹库范围；

确认码=18H 表示写 FLASH 出错；

sum 指校验和。

3.3.8 读出模板 PS_LoadChar

- 功能说明： 将 flash 数据库中指定 ID 号的指纹模板读入到特征文件缓冲区中。
- 输入参数： BufferID（默认为 2），PageID（指纹库模板号）
- 返回参数： 确认字
- 指令代码： 07H
- 指令包格式：

表 3-19 读出模板指令包格式

包头	芯片地址	包标识	包长度	指令码	缓冲区号	页码	校验和
2 bytes	4bytes	1 byte	2 bytes	1 byte	1 byte	2 bytes	2 bytes
0xEF01	xxxx	01H	0006H	07H	BufferID	PageID	sum

注： BufferID 默认为 2。

- 应答包格式：

表 3-20 读出模板指令应答包格式

包头	芯片地址	包标识	包长度	确认码	校验和
2 bytes	4bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	0003H	xxH	sum

注：确认码=00H 表示读出成功；
 确认码=01H 表示收包有错；
 确认码=0cH 表示读出有错或模板无效；
 确认码=0bH 表示 PageID 超出指纹库范围；
 sum 指校验和。

3.3.9 上传特征或模板 PS_UpChar

- 功能说明： 将特征缓冲区中的特征文件上传给上位机。
- 输入参数： BufferID（默认值）
- 返回参数： 确认字
- 指令代码： 08H
- 指令包格式：

表 3-21 上传特征或模板指令包格式

包头	芯片地址	包标识	包长度	指令码	缓冲区号	校验和
2 bytes	4bytes	1 byte	2 bytes	1 byte	1 byte	2 bytes
0xEF01	xxxx	01H	0004H	08H	BufferID	sum

注：从传感器中采集生成的特征，上传时 BufferID 默认为 1；从 Flash 中加载的特征，上传时 BufferID 默认为 2。

- 应答包格式：

表 3-22 上传特征或模板指令应答包格式

包头	芯片地址	包标识	包长度	确认码	校验和
2 bytes	4bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	0003H	xxH	sum

注：确认码=00H 表示随后发数据包；
 确认码=01H 表示收包有错；
 确认码=0dH 表示指令执行失败；
 sum 指校验和。

- 应答之后发送后续数据包。

表 3-23 UART 上传特征或模板数据包格式

包头	芯片地址	包标识	包长度	数据	校验和
2 bytes	4bytes	1 byte	2 bytes	N byte	2 bytes
0xEF01	xxxx	xxH	xxH	xxH	sum

注：包标识=02：数据包，且有后续包。

包标识=08：最后一个数据包，即结束包。

UART 上传特征或模板数据包时，按照预先设置的长度分包发送。

USB 上传特征或模板数据包时，直接发送整包数据，没有包头、芯片地址、包标识、包长度和校验和。

3.3.10 下载特征或模板 PS_DownChar

- 功能说明： 上位机下载特征文件到模块的一个特征缓冲区。
- 输入参数： BufferID（默认为 1）
- 返回参数： 确认字
- 指令代码： 09H
- 指令包格式：

表 3-24 下载特征或模板指令包格式

包头	芯片地址	包标识	包长度	指令码	缓冲区号	校验和
2 bytes	4bytes	1 byte	2 bytes	1 byte	1 byte	2 bytes
0xEF01	xxxx	01H	0004H	09H	BufferID	sum

注： BufferID 默认为 1。

- 应答包格式：

表 3-25 下载特征或模板指令应答包格式

包头	芯片地址	包标识	包长度	确认码	校验和
2 bytes	4bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	0003H	xxH	sum

注：确认码=00H 表示可以接收后续数据包；

确认码=01H 表示收包有错；

确认码=0eH 表示不能接收后续数据包；

sum 指校验和。

- 应答之后接收后续数据包。

表 3-26 UART 下载特征或模板数据包格式

包头	芯片地址	包标识	包长度	数据	校验和
2 bytes	4bytes	1 byte	2 bytes	N byte	2 bytes
0xEF01	xxxx	xxH	xxH	xxH	sum

注：包标识=02：数据包，且有后续包。

包标识=08：最后一个数据包，即结束包。

UART 下载特征或模板数据包时，按照预先设置的长度分包接收。

USB 下传特征或模板数据包时，直接接收整包数据，没有包头、芯片地址、包标识、包

长度和校验和。

3.3.11 上传图像 PS_Uplmage

- 功能说明： 将图像缓冲区中的数据上传给上位机。
- 输入参数： none
- 返回参数： 确认字
- 指令代码： 0aH
- 指令包格式：

表 3-27 上传图像指令包格式

包头	芯片地址	包标识	包长度	指令码	校验和
2 bytes	4bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	01H	0003H	0aH	000eH

- 应答包格式：

表 3-28 上传图像指令应答包格式

包头	芯片地址	包标识	包长度	确认码	校验和
2 bytes	4bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	0003H	xxH	sum

注：确认码=00H 表示接着发送后续数据包；
确认码=01H 表示收包有错；
确认码=0fH 表示不能发送后续数据包；
sum 指校验和。

- 应答之后发送后续数据包。

表 3-29 UART 上传图像数据包格式

包头	芯片地址	包标识	包长度	数据	校验和
2 bytes	4bytes	1 byte	2 bytes	N byte	2 bytes
0xEF01	xxxx	xxH	xxH	xxH	sum

注：包标识=02：数据包，且有后续包。

包标识=08：最后一个数据包，即结束包。

UART 上传图像数据包时，按照预先设置的长度分包发送。

USB 上传图像数据包时，直接发送整包数据，没有包头、芯片地址、包标识、包长度和校验和。

- 一个字节含两个像素，每个像素占 4bits。

6.3.12 下载图像 PS_DownImage

- 功能说明： 上位机下载图像数据给模块。

- 输入参数: none
- 返回参数: 确认字
- 指令代码: 0bH
- 指令包格式:

表 3-30 下载图像指令包格式

包头	芯片地址	包标识	包长度	指令码	校验和
2 bytes	4bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	01H	0003H	0bH	000fH

- 应答包格式:

表 3-31 下载图像指令应答包格式

包头	芯片地址	包标识	包长度	确认码	校验和
2 bytes	4bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	0003H	xxH	sum

注: 确认码=00H 表示可以接收后续数据包;

确认码=01H 表示收包有错;

确认码=0eH 表示不能接收后续数据包;

sum 指校验和。

- 应答之后接收后续数据包。

表 3-32 UART 下载图像数据包格式

包头	芯片地址	包标识	包长度	数据	校验和
2 bytes	4bytes	1 byte	2 bytes	N byte	2 bytes
0xEF01	xxxx	xxH	xxH	xxH	sum

注: 包标识=02: 数据包, 且有后续包。

包标识=08: 最后一个数据包, 即结束包。

UART 下载图像数据包时, 按照预先设置的长度分包接收。

USB 下载图像数据包时, 直接接收整包数据, 没有包头、芯片地址、包标识、包长度和校验和。

- 一个字节含两个像素, 每个像素占 4bits。

3.3.13 写系统寄存器 PS_WriteReg

- 功能说明: 写模块寄存器。
- 输入参数: 寄存器序号
- 返回参数: 确认字
- 指令代码: 0eH
- 指令包格式:

表 3-33 写系统寄存器指令包格式

包头	芯片地址	包标识	包长度	指令码	寄存器序号	内容	校验和
2 bytes	4bytes	1 byte	2 bytes	1 byte	1byte	1byte	2 bytes
0xEF01	xxxx	01H	0005H	0eH	4/5/6	xx	sum

- 应答包格式:

表 3-34 写系统寄存器指令应答包格式

包头	芯片地址	包标识	包长度	确认码	校验和
2 bytes	4bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	0003H	xxH	sum

注 1: 确认码=00H 表示 OK;

确认码=01H 表示收包有错;

确认码=1aH 表示寄存器序号有误;

sum 指校验和。

注 2: 写系统寄存器 (PS_WriteReg) 指令执行时, 先按照原配置进行应答, 应答之后修改系统设置, 并将配置记录于 FLASH, 系统下次上电后, 将按照新的配置工作。

表 3-35 寄存器配置表

寄存器号	寄存器名称	内容说明
4	波特率控制寄存器	9600 的倍数 N
5	比对阈值寄存器	1: level1 2: level2 3: level3 4: level4 5: level5
6	包大小寄存器	0: 32bytes 1: 64bytes 2: 128bytes 3: 256bytes

3.3.14 读系统基本参数 PS_ReadSysPara

- 功能说明:
 - ◆ 读取模块的基本参数 (波特率, 包大小等)。
 - ◆ 参数表前 16 个字节存放了模块的基本通讯和配置信息, 称为模块的基本参数。
- 输入参数: none
- 返回参数: 确认字+基本参数 (16bytes)
- 指令代码: 0fH
- 指令包格式:

表 3-36 读系统基本参数指令包格式

包头	芯片地址	包标识	包长度	指令码	校验和
2 bytes	4bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	01H	0003H	0fH	0013H

- 应答包格式:

表 3-37 读系统基本参数指令应答包格式

包头	芯片地址	包标识	包长度	确认码	基本参数列表	校验和
2 bytes	4bytes	1 byte	2 bytes	1 byte	16 bytes	2 bytes
0xEF01	xxxx	07H	13H	xxH	结构见表 6-38	sum

注：确认码=00H 表示 OK；

确认码=01H 表示收包有错；

sum 指校验和。

表 3-38 系统基本参数列表

名称	内容说明	偏移量 (字)	大小 (字)
状态寄存器	系统的状态寄存器内容	0	1
传感器类型	传感器类型代码。 ZN-630	1	1
指纹库大小	指纹库容量	2	1
安全等级	安全等级代码 (1/2/3/4/5)	3	1
设备地址	32 位设备地址	4	2
数据包大小	数据包大小代码： 0: 32bytes 1: 62bytes 2: 128bytes 3: 256bytes	6	1
波特率设置	N (波特率为 9600*N bps)	7	1

3.3.15 采样随机数 PS_GetRandomCode

- 功能说明：令芯片生成一个随机数并返回给上位机。
- 输入参数：none
- 返回参数：确认字
- 指令代码：14H
- 指令包格式:

表 3-39 采样随机数指令包格式

包头	芯片地址	包标识	包长度	指令码	校验和
2 bytes	4bytes	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	01H	0003H	14H	0018H

- 应答包格式:

表 3-40 采样随机数指令应答包格式

包头	芯片地址	包标识	包长度	确认码	随机数	校验和
2 bytes	4bytes	1 byte	2 bytes	1 byte	4 bytes	2 bytes
0xEF01	xxxx	07H	0007H	xxH	xxxx	sum

注：确认码=00H 表示生成成功；
 确认码=01H 表示收包有错；
 确认码=19H 表示随机数生成失败；
 sum 指校验和。

3.3.16 设置芯片地址 PS_SetChipAddr

- 功能说明： 设置芯片地址。
- 输入参数： none
- 返回参数： 确认字
- 指令代码： 15H
- 指令包格式：

表 3-41 设置芯片地址指令包格式

包头	芯片地址	包标识	包长度	指令码	芯片地址	校验和
2 bytes	4bytes	1 byte	2 bytes	1 byte	4 bytes	2 bytes
0xEF01	xxxx	01H	0007H	15H	xxxx	sum

- 应答包格式：

表 3-42 设置芯片地址指令应答包格式

包头	芯片地址	包标识	包长度	确认码	校验和
2 bytes	4 byte	1 byte	2 bytes	1 byte	2 bytes
0xEF01	xxxx	07H	0007H	xxH	sum

注：确认码=00H 表示生成地址成功；
 确认码=01H 表示收包有错；
 sum 指校验和。

- 上位机下传指令包时芯片地址采用缺省地址：0xffffffff，应答包的地址域即采用新生成的地址。
- 本指令执行后，芯片地址随即固定下来，保持不变。只有清空 FLASH 才能改变芯片地址。
- 本指令执行后，所有数据包都得用该生成的地址。

3.3.17 读 flash 信息页 PS_ReadINFpage

- 功能说明： 读取 FLASH Information Page 所在的信息页（512bytes）。
- 输入参数： none