

用户软件开发手册

目录

声明	1
联系方式.....	2
版本历史	3
目录	4
一、 软件通信协议对接.....	6
(一) 通信协议.....	6
1. 通信消息格式.....	6
2. 人脸锁算法模组消息接收(H>>M).....	7
3. 人脸锁算法模组消息发送(M>>H).....	16
(二) 功能实现示例.....	26
1. 主控接收消息流程.....	26
2. 一般消息处理流程.....	26
3. 上下电流程	27
4. 录入人脸流程.....	27
5. 录入说明	28
6. 单帧录入.....	29
7. 验证人脸流程.....	30
8. 加密通信流程.....	30
9. 照片下发注册示例.....	31
MID_ENROLL_WITH_PHOTO.....	31

(三) 补充说明.....	34
1. MID_REPLY.....	34
2. MID_NOTE	35
3. MID_RESET	36
4. MID_GETSTATUS.....	36
5. MID_VERIFY	36
6. MID_ENROLL/MID_ENROLL_SINGLE	37
7. MID_ENROLL_ITG.....	38
8. MID_DELUSER.....	39
9. MID_DELALL	39
10. MID_GETUSERINFO	39
11. MID_ENROLL_WITH_PHOTO.....	39
12. MID_DEMOMODE.....	40
13. MID_GET_ALL_USERID	40

一、 软件通信协议对接

(一) 通信协议

人脸识别模组处于从属地位，主控需要通过不同指令让模组完成各种功能。

1. 通信消息格式

主控与双目人脸识别模组通信的基本消息格式如下表所示

基本消息格式				
SyncWord	MsgID	Size	Data	ParityCheck
2 bytes	1 byte	2 bytes	N bytes	1 byte

下表是对各个字段的详细说明。

基本消息格式字段详细说明		
字段	长度	说明
SyncWord	2bytes	固定的消息开头同步字 0xEF 0xAA
MsgID	1byte	消息 ID (例如 RESET)
Size	2bytes	Data size, 单位 byte
Data	N bytes	消息对应的 data,command 消息对应的参数。65535>=N>=0, N=0 表示此消息无参数。
Parity Check	1 byte	协议的奇偶校验码，计算方式为整条协议除去 SyncWord 部分后，其余字节按位做 XOR 运算。

2. 人脸锁算法模组消息接收(H>>M)

模组接收到的完整协议格式如下表所示，主控应按照该格式向模组发送命令。

H>>M 协议格式说明

名称	SyncWor	MsgID	Size		Data	ParityCheck
字节数	2 bytes	1 byte	2 bytes		N bytes	1 byte
内容	0xEF AA	command	高八位	低八位	data	checksum

command 和data 定义如下表所示。

H>>M 协议格式数据详细说明

command (*表示携带 data)	code	data		说明
		结构体	内容	
RESET	0x10	无		无
GET STATUS	0x11	无		无
VERIFY	0x12	s_msg_verify_data	pd_rightaway (1byte)	保留 (默认:0)
			Timeout (1byte)	解锁超时时间 (单位 s)
AUTO_VERIFY* (AI-10模组适用)	0x12	s_msg_Auto_verify_data	at_verify (1byte)	启动低功耗自动探测 (yes:1 no:0) 1=当判断到人脸或二维码后自动进行识别并输出结果；（注：掌静脉识别只在探测到人脸后才启动） 0=退出该模式。 发送0x10 RESET命令，也可以直接退出该模式。
			Timeout (1byte)	单次识别超时时间 (单位 s) 检测到有人脸经过，自动开始识别，此时若人已走开则xS后退出识别，模组继续进行低功耗人脸探测

VERIFY* (FM226T-XF系列及 AI-10模组适用)	0x12	s_msg_verify_data	pd_rightaway (1byte)	保留 (默认:0)
			Timeout (1byte)	解锁超时时间 (单位 s)
			Max_Recognition_times (1byte) (可选)	重复识别次数限定, 范围从0-255,这个数值越小, 则尝试识别的次数越少, 返回未知用户的时间就越快! 具体时间由用户来测试获得。此参数为扩展参数, 如下发时不发送此参数, 则模组默认尝试识别次数为30。
ENROLL	0x13	s_msg_enroll_data	admin	是否设置为管理员(yes:1 no:0)
			user_name (32bytes)	录入用户姓名
			s_face_dir (1byte)	用户需要录入的方向, 具体参数下表 “人脸方向定义” 所示
			timeout (1 byte)	录入超时时间 (单位 s)
ENROLL_SINGLE	0x1D	s_msg_enroll_data	admin	是否设置为管理员(yes:1 no:0)
			user_name (32bytes)	录入用户姓名
			s_face_dir (1byte)	不使用
			timeout (1 byte)	录入超时时间 (单位 s)
DELETE USER*	0x20	s_msg_deluser_data	user_id_heb (1byte) user_id_leb (1byte)	待删除用户的ID
DELETE ALL	0x21	无		删除所有用户
GET USER INFO*	0x22	s_msg_getuser_info_data	user_id_heb (1byte)	已注册用户的ID
			user_id_leb	

			(1byte)	
FACE RESET	0x23	无		清除录入状态
MID_GET_ALL_U SERID	0x24	00(1byte)		获取所有已注册用户的 数量和ID
MID_ENROLL_ITG	0x26	s_msg_enroll_ itg	admin	是否设置为管理员(yes:1 no:0)
			user_name (32bytes)	录入用户姓名
			face_dir (1byte)	不使用
			enroll_type (1 byte)	注册类型: 交互 录入或单帧录入
			enable_duplica te (1 byte)	0
			timeout (1 byte)	录入超时时间 (单位 s)
			Reserved (3 byte)	预留位
GET VERSION	0x30	无		获取软件版本
INIT ENCRYPTIO*	0x50	s_msg_init_encr yption_data	seed[4](4 bytes)	主控发送一个随机数
			mode	
MID_SET_RELE A SE_ENC_KEY	0x52	s_msg_enc_ke y_number_dat a	enc_key_number [16]	加密序列
MID_SET_DEBU G_ENC_KEY	0x53	s_msg_enc_ke y_number_dat a	enc_key_num ber[16]	加密序列
MID_GET_SN	0x93	无		设备唯一序列号
READ_USB_UVC_PA RAMETERS	0xB0	无		usb 传图参数设置
SET_USB_UVC_PAR AMETERS	0xB1	s_msg_us b_uvc_data	USB type (1byte)	USB1.1 : 0x11 USB2.0 : 0x20

(FM223系列 & FM888系列适用)			图像旋转180 度和 图像镜像 (1byte)	BIT0 : 0-不旋转, 1-旋转 180度 BIT1 : 0-不镜像, 1-镜像 翻转
			JPEG 图 像 质 量百分比(1byte)	10 ~ 99
SET_USB_UVC_PAR AMETERS (FM226系列、AI-10 模组适用)	0xB1	s_msg_us b_uvc_data	USB Type (1byte)	USB1.1: 0x11 USB2.0: 0x20 BIT7:0-Bulk 1-ISOC (注: 0xa0-USB2.0&ISOC 0x20-USB2.0&Bulk 0x11-USB1.1&Bulk 0x91-USB1.1&ISOC)
			UVC码率 (1byte)	Uvc的码率, 单位Mbps, 例: 0x18 = 24Mbps
			MJPG的质量 (1byte)	10-100 , 如0x4b = 75 意味着75%的质量, 数值 越接近100 , 图片越清 晰, 但会受到码率控制。 因此高质量图片, 需要 对应设置高码率才可以!
			图像属性 (1byte)	BIT0: 1 启用镜像 BIT1: 1 启用倒转180度
MID_SCAN_QR_CO DE	0x70	s_msg_sca n_QR	00 (1 byte)	读取二维码信息
			timeout (1 byte)	超时时间 (单位: s)
MID_SNAP&UPLOA D_IMAGE_1	0x71	s_msg_sna pimage	00(1byte)	抓拍照片并上传
MID_SNAP&UPLOA D_IMAGE_2	0x71	s_msg_sna pimage	Seq(1byte)	上传照片 (240*320pixel) Seq累加1 (Seq从1开始 接收照片数据), 每一包 固定传送1024字节照片 数据, 最后一包不足 1024字节时按实际剩余 字节数传输。
MID_SNAP&UPLOA D_FACEIMAGE_1	0x72	s_msg_sna pfaceimage	00(1byte)	抓拍人脸照片后本地比 对, 完成上传。

				判断到合格到正脸照片后，自动抓拍、本地比对并上传照片。人脸照片为320*320pixel。
MID_SNAP&UPLOAD_FACEIMAGE_2	0x72	s_msg_snapfaceimage	Seq(1byte)	上传人脸照片 Seq累加1（Seq从1开始接收照片数据），每一包固定传送1024字节照片数据，最后一包不足1024字节时按实际剩余字节数传输。
ENROLL_SNAPFACE IMAGE	0x73	s_msg_enroll_data	admin	是否设置为管理员(yes:1 no:0)
			user_name (32bytes)	录入用户姓名
			s_face_dir(1byte)	预留
			timeout (1 byte)	预留
MID_SNAP&UPLOAD_IMAGE_1	0x74	s_msg_snapimage_b	Seq (1byte)	抓拍全视角大图照片，Seq=0时，DPI有效
			DPI (1byte)	图片分辨率 0 - 480x640 1 - 600x800 2 - 1200x1600 (仅AI-10有效)
MID_SNAP&UPLOAD_IMAGE_2	0x74	s_msg_snapimage_b	Seq(1byte)	当Seq>0时，上传照片 Seq累加1（Seq从1开始接收照片数据），每一包固定传送1024字节照片数据，最后一包不足1024字节时按实际剩余字节数传输。
MID_ENROLL_WITH	0xD7	s_msg_enroll_photo&id	00(1byte)	下发照片或特征注册 (第一包命令)
			00(1byte)	
			Phpoto len (4bytes) (大端模式)	照片或特征长度

_PHOTO&ID_1 (建议优先使用 0xF7 命令)			BioType (1byte) 0为普通照片模式, 1为加密照片模式, 2 为普通特征码模式 (2048字节[彩色]或4096[彩色+红外]), 3 为压缩特征码模式 (1024字节)	特征文件为4100或2052或1028字节, 其中4096或2048或1024字节是特征码, 4个字节的crc32 (大端) CRC32的初始向量值是0xffffffff
			UID (1byte) 1-100 (FM226) 或UID(2byte) 1-1000 (AI-10)	开发者指定的用户ID。 UID=0 代表由模组生成ID, FM226设置成1-100则是用户指定ID, 其他值会返回失败。
			Duplicate_CkFg (1byte)	重复录入检查标志。 设置为1, 则启动检查, 会在当前已有的数据库中查询, 如果与当前下发的用户ID (已存在的用户) 比对成功, 则允许下发覆盖。如果与其他用户id重复, 则返回失败。 设置为0则不进行重复录入检查。
			NameLen (1byte)	姓名长度, 不大于20, 当为0或无该字段时默认无姓名【可选】
			Name String	用户姓名内容
MID_ENROLL_WITH _PHOTO&ID_2	0xD7	s_msg_enroll_ photo&id	Seq heb (1byte)	当Seq>0时, 照片或特征数据传输Seq累加1 (Seq从1开始发送照片数据) hoto data为照片数据 照片数据分包大小
			Seq leb (1byte)	
			Phphoto data[n] (n bytes)	

				MTU≤246
MID_UPGRADE_FW	0xF6	无		启动通过U盘 升级模组固件
MID_ENROLL_WITH _PHOTO_1	0xF7	s_msg_enroll_ photo	00(1byte)	下发照片或特征注册 (第一包命令) 其中特征文件为 4100 或 2052或1028字节, 其中 4096 或2048或1024字 节是特征码, 4个字节 的 crc32 (大端) CRC32的 初始向量值是0xffffffff
			00(1byte)	
			Phpoto len (4bytes) (大端模 式)	
			BioType(1byte) 0为普通照片模式, 1为加密照片模式, 2 为普通特征码模 式 (2048字节[彩 色]或4096[彩色+ 红外]) , 3 为压缩特征码模 式 (1024字节)	
		扩展参数区: 如不需要设置姓 名, 可不发扩展 参数, 当模组收 到的参数总长度 超过7时, 认为 存在扩展参数!	Name Len(1byte) (可选)	姓名字符串的长度1-20 超过20会按照20来保存
			Name String[n] (可选)	姓名字符串 n<20
MID_ENROLL_WITH _PHOTO_2	0xF7	s_msg_enroll_ photo	Seq heb (1byte)	照片或特征数据传输 Seq累加1 (Seq从1开始 发送照片数据), hoto data为照片数据, 照片数 据分包大小MTU≤246
			Seq leb (1byte)	
			Phpoto data[n] (n bytes)	
MID_READ_FEATUR E_1 (AI-10模组适用)	0xFA	s_msg_read_fe ature_data_1	seq_id_heb (1byte)	包序号 (00 00)
			seq_id_leb (1byte)	
			user_id_heb (1byte)	要读取的用户ID (1~1000为人脸用户,

			user_id_leb (1byte)	1001~2000为掌纹掌静脉)
			feature_type (1byte)	1: 彩色人脸模板 2: 彩色+外红人脸模板 3: 掌纹掌静脉模板
MID_READ_FEATURE_2 (AI-10模组适用)	0xFA	s_msg_read_feature_data_2	seq_id_heb (1byte)	包序号 (每包246byte, 计算出总包数, 然后从1开始发送依次累加, 到最后一包数据接收完)
			seq_id_leb (1byte)	
			user_id_heb (1byte)	要读取的用户ID (1~1000为人脸用户, 1001~2000为掌纹掌静脉)
			user_id_leb (1byte)	
			Feature data[n] (nbyte)	接收模板数据 (每包246byte, 最后一包为剩余实际长度数据)
MID_WRITE_FEATURE_1 (AI-10模组适用)	0xFB	s_msg_write_feature_data_1	seq_id_heb (1byte)	包序号 (00 00)
			seq_id_leb (1byte)	
			feature_type (1byte)	1: 彩色人脸模板 2: 彩色+外红人脸模板 3: 掌纹掌静脉模板
		扩展参数区: 如不需要设置姓名, 可不发扩展参数, 当模组收到的参数总长度超过3时, 认为存在扩展参数!	Name Len(1byte) (可选)	姓名字符串的长度1-20 超过20会按照20来保存
			Name String[n] (可选)	姓名字符串 n<20
MID_WRITE_FEATURE_2 (AI-10模组适用)	0xFB	s_msg_write_feature_data_2	seq_id_heb (1byte)	包序号 (每包246byte, 计算出总包数, 然后从1开始发送依次累加, 到最后一包
			seq_id_leb	

			(1byte)	数据发送完)
			feature_type (1byte)	1: 彩色人脸模板 2: 彩色+外红人脸模板 3: 掌纹掌静脉模板
			Feature data[n] (n byte)	模板数据 (每包246byte, 最后一包为剩余实际长度数据)
MID_DUPLICATE_CHECK	0xFC	s_msg_duplicate_check	checkflag	0:不查重 1: 查重 (下发照片注册和下发特征模式时是否查重)
DEMO MODE*	0xFE	s_msg_demo_mode_data	enable (1byte)	enable:1 disable: 0

人脸方向定义如下表所示。

人脸方向定义说明

st_face_dir(人脸方向定义)	code	说明
FACE_DIRECTION_UP	0x10	录入朝上的人脸
FACE_DIRECTION_DOWN	0x08	录入朝下的人脸
FACE_DIRECTION_LEFT	0x04	录入朝左的人脸
FACE_DIRECTION_RIGHT	0x02	录入朝右的人脸
FACE_DIRECTION_MIDDLE	0x01	录入正向的人脸
FACE_DIRECTION_UNDEFINE	0x00	未定义, 默认为正向

人脸录入命令类型定义如下表所示。

人脸录入命令类型定义说明

enroll_type(人脸录入命令类型)	Code	说明
FACE_ENROLL_TYPE_INTERACTIVE	0x0	交互录入
FACE_ENROLL_TYPE_SINGLE	0x1	单帧录入

示例: 0xEF 0xAA 0x10 0x00 0x00 0x10

H>>M 协议示例

名称	SyncWord	MsgID	Size		Data	ParityCheck
内容	0xEFAA	0x10 (MID_RESET)	0x00	0x00	无	0x10

这条消息是主控发送给模组的 RESET 消息，数据为长度为 0。

3. 人脸锁算法模组消息发送(M>>H)

模组主要发送三种类型的消息，分别是 REPLY，NOTE，IMAGE。

1) REPLY 消息的发送

模组向主控发送的 REPLY 消息的完整协议如下表所示。

M>>H REPLY 消息协议格式说明

名称	SyncWord	MsgID	Size		Data			ParityCheck
字节数	2 bytes	1 byte	2 bytes		N bytes			1 byte
内容	0xEFAA	MID_REPLY (0x00)	高八位	低八位	s_msg_reply_data			checksum
					mid (1byte)	result (1byte)	data[0] (n-byte)	

mid 表示模组当前正在处理的任務，例如当 mid = MID_ENROLL(0x13) 时，表示该消息是模组处理完 enroll 任务后回复的消息。mid 详细如下表所示。

M>>H REPLY 消息或 ImageOrTemplate 消息中 mid 定义说明

mid (*表示携带 data)	code	data[0]		说明
		结构体	内容	
MID_RESET	0x10	无		无
				模组的状态包括：

MID_GETSTATUS*	0x11	s_msg_repy_getstatus_data	status (1byte)	IDLE(0),BUSY(1),ERROR(2),INVALID(3)
MID_VERIFY* (只有当 result 结果为 MR_SUCCESS 时才有 user_id)	0x12	s_msg_reply_verify_data	user_id_heb (1byte)	已验证用户的 ID
			user_id_leb (1byte)	
			user_name (32 bytes)	用户名字
			admin (1byte)	是否为管理员
			unlockStatus (1 byte)	无
MID_ENROLL* (只有当 result 结果为 MR_SUCCESS 时才有 user_id)	0x13	s_msg_reply_enroll_data	user_id_heb (1byte)	已注册用户的 ID
			user_id_leb (1byte)	
			face_direction(1 byte)	各个方向人脸的录入状态
MID_ENROLL_SINGLE* (只有当 result 结果为 MR_SUCCESS 时才有 user_id)	0x1D	s_msg_reply_enroll_single_data	user_id_heb (1byte)	已注册用户的 ID
			user_id_leb (1byte)	
			face_direction(1 byte)	01 (表示正脸录入)
MID_DELUSER	0x20	无		无
MID_DELALL	0x21	无		无
MID_GETUSERINFO*	0x22	s_msg_reply_getuserinfo_data	user_id_heb (1byte)	已注册用户的 ID
			user_id_leb (1byte)	
			user_name (32bytes)	已注册用户的姓名
			admin(1byte)	是否为管理员
MID_FACERESET	0x23	无		无
MID_GET_ALL_USERID*	0x24	s_msg_reply_all_userid_data	user_counts (1 byte)	已注册用户数量, 单包一次最多为100

				个, 当返回的数量小于100时结束。
			users_id[MAX_USER_COUNTS*2] (≤100*2 bytes)	所有已注册用户ID, 使用连续两个字节存储一个ID, 先存高八位
MID_ENROLL_ITG*(只有当 result 结果为 MR_SUCCESS 时才有 user_id)	0x26	s_msg_reply_enroll_data	user_id_heb (1byte) user_id_leb (1byte)	已注册用户的 ID
MID_GET_VERSION*	0x30	s_msg_reply_version_data		版本信息
MID_INIT_ENCRYPTION	0x50	s_msg_reply_init_encryption_data	device_id[20](20 bytes)	设备 id 信息
MID_SET_RELEASE_ENC_KEY	0x52	无		无
MID_SET_DEBUG_ENC_KEY	0x53	无		无
MID_GET_SN	0x93	Device_SN[32] (32bytes)		设备唯一序列号信息, 前8字节有效
READ_USB_UVC_PARAMETERS (FM223系列 & FM888系列适用)	0xB0	s_msg_usb_uvc_data	USB type (1byte)	USB1.1 : 0x11 USB2.0 : 0x20
			图像旋转180度和图像镜像 (1byte)	BIT0 : 0-不旋转, 1-旋转180度 BIT1 : 0-不镜像, 1-镜像翻转
			JPEG图像质量百分比(1byte)	10 ~ 99
READ_USB_UVC_PARAMETERS (FM226系列适用)	0xB0	s_msg_usb_uvc_data	USB Type (1byte)	USB1.1: 0x11 USB2.0: 0x20 BIT7:0-Bulk 1-ISOC (0xa0-USB2.0&ISOC 0x20-USB2.0&Bulk 0x11-USB1.1&Bulk 0x91-USB1.1&ISOC)
			UVC码率	Uvc 的码率, 单位

			(1byte)	Mbps, 例: 0x18 = 24Mbps
			MJPG的质量 (1byte)	10-100, 当前图像质量
			图像属性 (1byte)	BIT0: 1 启用镜像 BIT1: 1 启用倒转 180度
SET_USB_UVC_PARAMETERS	0xB1	无		Usb 传图参数设置结果
MID_SNAP&UPLOAD_IMAGE_1	0x71	s_msg_reply_snapimage1	result(1byte)	0=成功, 其他=失败
			Phpoto len (4bytes) (大端模式)	如00 00 28 a3 代表图片长度, 大端模式 等于0x28a3
MID_SNAP&UPLOAD_IMAGE_2	0x71	s_msg_reply_snapimage2 (注: 此应答按照 ImageOrTemplate 消息协议格式 MsgID Size Data 返回)	Phpoto data[n] (n bytes)	Size为本包照片数据长度, 每一包固定传送1024字节照片数据, 最后一包不足1024字节时按实际剩余字节数传输。 Photo data为照片数据
MID_SNAP&UPLOAD_FACEIMAGE_1	0x72	s_msg_reply_snapfaceimage1	SnapFaceresult (1byte)	0=成功, 其他=失败
			Phpoto len (4bytes) (大端模式)	如00 00 28 a3 代表图片长度, 大端模式 等于0x28a3
			UserID (2bytes) (大端模式)	搜索本地用户ID。 UserID等于00 00, 代表模组内没有对应用户, 如果非0则代表有对应用户, 大端模式 id_high id_low

MID_SNAP&UPL OAD_FACEIMAGE_2	0x72	s_msg_reply_snapfaceimage2 (注: 此应答按照 ImageOrTemplate 消息协议格式 MsgID Size Data 返回)	Phpoto data[n] (n bytes)	Size为本包照片数据长度, 每一包固定 传送1024字节照片 数据, 最后一包不足 1024字节时按实际 剩余字节数传输。 Photo data为照片 数据
ENROLL_SNAPFACEIMAGE* (只有当 SnapFacerestult 结 果为 0 时才有 user_id)	0x73	s_msg_reply_enrollsnapfaceimage_data	user_id_heb (1byte)	已注册用户的 ID 注: 配合0x72未搜索 到本地用户时使用, 该录入命令不查重。
			user_id_leb (1byte)	
			face_direction(1 byte)	默认为0
MID_SNAP&UPL OAD_IMAGE_1	0x74	s_msg_reply_snapimage1	result(1byte)	0=成功, 其他=失败
			Phpoto len (4bytes) (大端模 式)	如00 00 28 a3 代 表图片长度, 大端模 式 等于0x28a3
MID_SNAP&UPL OAD_IMAGE_2	0x74	s_msg_reply_snapimage2 (注: 此应答按照 ImageOrTemplate 消息协议格式 MsgID Size Data 返回)	Phpoto data[n] (n bytes)	Size为本包照片数据长度, 每一包固定 传送1024字节照片 数据, 最后一包不足 1024字节时按实际 剩余字节数传输。 Photo data为照片 数据
MID_UPGRADE_FW	0xF6	s_msg_reply_upgradefw	Progress (1byte)	升级进度百分比
MID_ENROLL_WITH_PHOTO_1	0xF7	s_msg_reply_enroll_photo_1	Seq heb(1 byte)	包序号
			Seq leb(1 byte)	
			UserID heb(1 byte)	保留 (待返回的用户ID, 默认为00 00)
			UserID leb(1 byte)	
MID_ENROLL_WITH_PHOTO_2	0xF7	s_msg_reply_enroll_photo_2	Seq heb(1 byte)	包序号
			Seq leb(1 byte)	

			UserID heb(1 byte)	注册成功时用户ID (仅在照片发送完成后, 注册成功才有对应UserID, 否则为0)
			UserID leb(1 byte)	
MID_ENROLL_WITH_PHOTO_F (建议优先使用0xF7命令)	0xD7	s_msg_reply_enroll_photo_f	Seq heb(1 byte)	包序号
			Seq leb(1 byte)	
			UserID heb(1 byte)	已注册的用户ID (仅在照片发送完成后, 注册成功才有对应UserID, 否则为0)
			UserID leb(1 byte)	
MID_READ_FEATURE_1	0xFA	s_msg_reply_read_feature_data	result(1 byte)	0=成功, 其他=失败
			Feature len (4bytes) (大端模式)	如00 00 16 D4 代表掌纹掌静脉特征长度, 大端模式等于0x16DE
MID_READ_FEATURE_2	0xFA	s_msg_reply_read_feature_data (注: 此应答按照ImageOrTemplate消息协议格式MsgID Size Data返回)	Feature data[n] (n bytes)	Size为本包特征数据长度, 每一包固定传送246字节特征数据, 最后一包不足246字节时按实际剩余字节数传输。Feature data为特征数据
MID_WRITE_FEATURE_1	0xFB	s_msg_reply_write_feature_data_1	Seq heb(1 byte)	包序号
			Seq leb(1 byte)	
			UserID heb (1 byte)	保留 (待返回的用户ID, 默认为00 00)
			UserID leb (1 byte)	
MID_WRITE_FEATURE_2	0xFB	s_msg_reply_write_feature_data_2	Seq heb(1 byte)	包序号
			Seq leb(1 byte)	

			UserID heb (1 byte) UserID leb (1 byte)	注册成功时UserID (仅在照片发送完成后, 注册成功才有对应UserID, 否则为0)
MID_DUPLICATE_CHECK	0xFC	无		无
MID_DEMOMODE	0xFE	无		无

result 表示该命令的执行结果, 详细如下表所示。

M>>H REPLY 消息中 result 定义说明

result	code	说明
MR_SUCCESS	0	成功
MR_REJECTED	1	模组拒绝该命令
MR_ABORTED	2	录入/验证算法已终止
MR_FAILED4_CAMERA	4	相机打开失败
MR_FAILED4_UNKNOWNREASON	5	未知错误
MR_FAILED4_INVALIDPARAM	6	无效的参数
MR_FAILED4_NOMEMORY	7	内存不足
MR_FAILED4_UNKNOWNUSER	8	没有已录入的用户
MR_FAILED4_MAXUSER	9	录入超过最大用户数量
MR_FAILED4_FACEENROLLED	10	人脸已录入
MR_FAILED4_LIVENESSCHECK	12	活体检测失败
MR_FAILED4_TIMEOUT	13	录入或解锁超时
MR_FAILED4_AUTHORIZATION	14	加密芯片授权失败
MR_FAILED4_READ_FILE	19	读文件失败
MR_FAILED4_WRITE_FILE	20	写文件失败
MR_FAILED4_NO_ENCRYPT	21	通信协议未加密
MR_FAILED4_NO_RGBIMAGE	23	RGB 图像没有 ready
MR_FAILED4_JPGPHOTO_LARGE	24	JPG照片过大 (照片注册)

MR_FAILED4_JPGPHOTO_SMALL	25	JPG照片过小 (照片注册)
---------------------------	----	----------------

示例: 0xEF 0xAA 0x00 0x00 0x05 0x13 0x00 0x00 0x03 0x1F 0x0A

M>>H REPLY 消息协议示例

名称	Sync Word	MsgID	Size		Data					Parity Check
内容	0xEFAA	0x00 (MID_REPLY)	0x00	0x05	s_msg_reply_data					0x0A
					0x13 (MID_ENROLL)	0x00 (MR_SUCESS)	0x00 (user_idheb)	0x03 (user_idleb)	0x1F (face_direction)	

这条消息表示这是模组回复给主控的 REPLY 消息, 数据部分占5 字节, 录入成功且录入的用户 ID 是3。

2) NOTE 消息的发送

NOTE 消息主要作用是主动向主控返回一些信息, 目前 NOTE 消息主要在三种情况下发送:

- 开机时模组向主控发送 NID_READY;
- 录入过程中模组向主控发送人脸信息;
- 解锁过程中模组向主控发送人脸信息。

模组向主控发送的NOTE 消息的完整协议如下表所示。

M>>H NOTE 消息协议格式说明

名称	SyncWord	MsgID	Size		Data		ParityCheck
字节数	2 bytes	1 byte	2 bytes		N bytes		1 byte
内容	0xEFAA	MID_NOTE (0x01)	高 八 位	低 八 位	s_msg_note_data		checksum
					nid (1byte)	data[0] (n-bytes)	

nid 表示enroll 过程中算法的执行结果, 详细如下:

M>>H NOTE 消息中 nid 定义说明

nid (*表示携带 data)	code	说明
NID_READY	0	模组已准备好
NID_FACE_STATE*	1	算法执行成功, 并且返回人脸信息_note_data_face

NID_UNKNOWNERROR	2	未知错误
NID_OTA_DONE*	3	无
NID_EYE_STATE	4	无

NID_FACE_STATE 所携带的数据 data 主要存储了人脸信息，具体如下：

M>>H NOTE 消息中 NID_FACE_STATE 对应的 Data 说明

结构	s_note_data_face							
内容	state	left	top	Right	bottom	yaw	pitch	roll
类型	int16_t	int16_t	int16_t	int16_t	int16_t	int16_t	int16_t	int16_t
字节	2bytes	2bytes	2bytes	2bytes	2bytes	2bytes	2bytes	2bytes

state:表示人脸当前的状态，主要包括以下几种情况：

M>>H NOTE 消息中 NID_FACE_STATE 对应的 Data 的 state 说明

人脸状态(state)	cod	说明
FACE_STATE_NORMAL	0	人脸正常
FACE_STATE_NOFACE	1	未检测到人脸
FACE_STATE_TOOUP	2	人脸太靠近图片上边沿，未能录入
FACE_STATE_TOODOWN	3	人脸太靠近图片下边沿，未能录入
FACE_STATE_TOOLEFT	4	人脸太靠近图片左边沿，未能录入
FACE_STATE_TOORIGHT	5	人脸太靠近图片右边沿，未能录入
FACE_STATE_FAR	6	人脸距离太远，未能录入
FACE_STATE_CLOSE	7	人脸距离太近，未能录入
FACE_STATE_EYEBROW_OCCLUSION	8	眉毛遮挡
FACE_STATE_EYE_OCCLUSION	9	眼睛遮挡
FACE_STATE_FACE_OCCLUSION	10	脸部遮挡
FACE_STATE_DIRECTION_ERROR	11	录入人脸方向错误
FACE_STATE_EYE_CLOSE_STATUS_OPEN_EYE	12	在闭眼模式检测到睁眼状态

FACE_STATE_EYE_CLOSE_STATUS	13	闭眼状态
FACE_STATE_EYE_CLOSE_UNKNOW_STATUS	14	在闭眼模式检测中无法判定睁闭眼状

s_note_data_face 中其它成员的具体含义如下所示：

left:人脸框距离图片最左侧的距离（负数表示人脸框已超出图片最左侧）

top:人脸框距离图片最上方的距离（负数表示人脸框已超出图片上方）

right:人脸框距离图片最右侧的距离（负数表示人脸框已超图片最右侧）

bottom:人脸框距离图片最下方的距离（负数表示人脸框已超出图片最下方）

yaw,pitch,roll 表示的旋转方向。其中 yaw 为负表示左转头，yaw 为正表示右转头；pitch 为负表示向上抬头，pitch 为正表示向下低头；roll 为负表示向右歪头，roll 为正表示向左歪头。

示例：0xEF 0xAA 0x01 0x00 0x01 0x00 0x00

M>>H NOTE 消息协议示例

名称	SyncWord	MsgID	Size		Data		ParityCheck
内容	0xEFAA	0x01(MID_NOTE)	0x00	0x01	s_msg_note_data		0x00
					0x00(NID_READY)	无	

这条消息表示这是模组向主控发送的 NOTE 消息，数据长度为 1 字节，模组已经准备好接收其他命令。

3) ImageOrTemplate 消息发送

模组向主控发送的 图像（Image）或特征模板（Template）消息的完整协议如下表所示。

M>>H ImageOrTemplate 消息协议格式说明

名称	SyncWord	MsgID	Size		Data	ParityCheck
字节数	2 bytes	1 byte	2 bytes		N bytes	1 byte
内容	0xEFAA	mid	高八	低八	image data	checksum

mid 表示模组当前正在处理的任务，例如当 mid = MID_SNAP&UPLOAD_FACEIMAGE_2(0x72) 时，

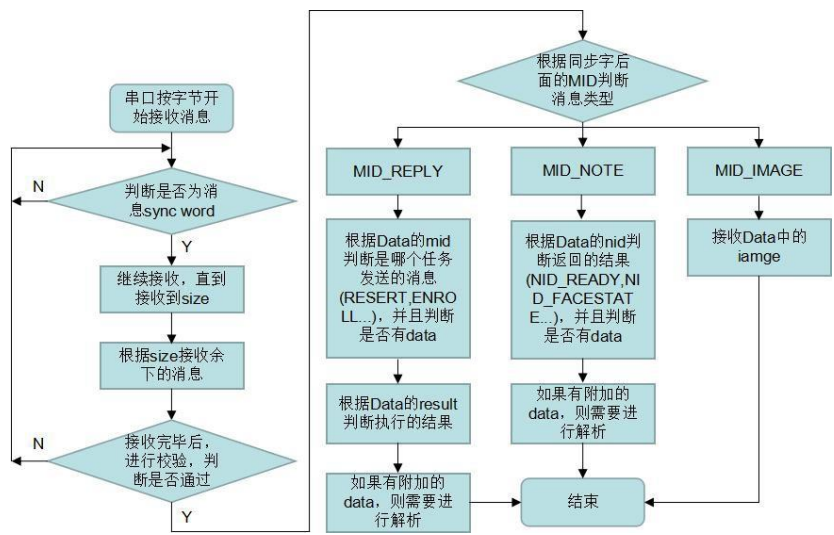
表示该消息是模组处理完 抓拍 任务后回复的消息。mid 详细内容见“**REPLY 消息或 ImageOrTemplate**

消息中 mid 定义说明”所示。

(二) 功能实现示例

1. 主控接收消息流程

主控接收模组发送的消息可以遵循下图所示流程。



主控接收模组消息流程图

2. 一般消息处理流程



一般消息处理流程图

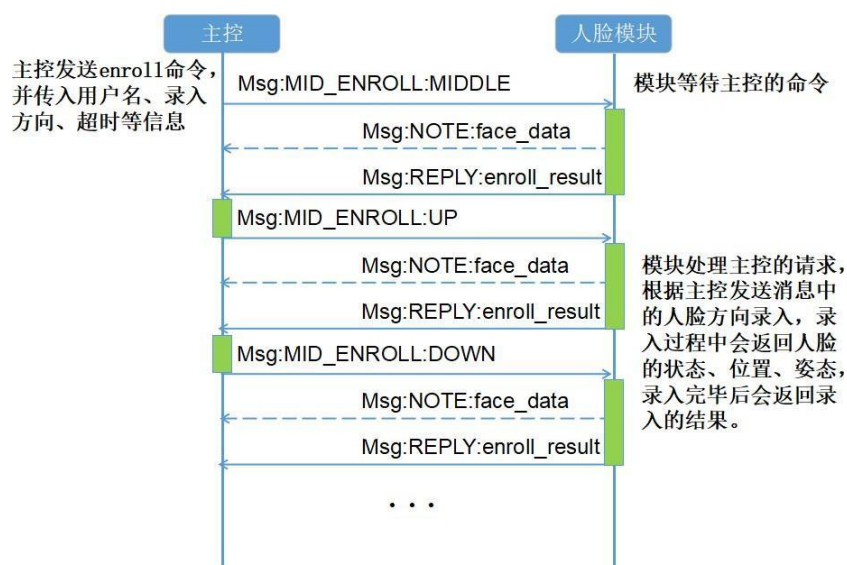
3. 上下电流程

主控主动控制模组的上电和下电，如下图所示。模组上电即进入启动流程，启动进入 standby 状态会通知主控 MSG:NOTE:READY，主控开始发送命令消息，模组处理并返回结果。模组无消息发送或超时，主控可断电。



上下电处理流程图

4. 录入人脸流程



录入人脸流程图

在录入过程中，人脸锁算法模组不限制录入方向的顺序，用户可以任意组合录入方向的顺序，上图是以其中一种顺序为例进行说明。

当主控下达录入某个方向的人脸的指令后，模组开始工作，并实时地返回人脸的状态、位置、姿态等

信息 (以NOTE 消息的形式返回), 用户可根据该信息提醒录入人进行姿态位置的调整, 当录入成功后, 模组将以 REPLY 消息的形式返回录入结果, 如果录入不成功, 模组也会相应地返回失败的原因。

值得注意的是, 当模组采集到的第一帧图像就录入成功时, 将直接以 REPLY 消息的形式返回结果。
录入过程中可通过 FACE RESET 指令终止录入, 先前的录入状态也会清零。

如果在录入过程中突然断电, 之前录入的人脸也不会保存。

5. 录入说明

人脸录入各个方向的角度说明如下:

人脸录入角度说明

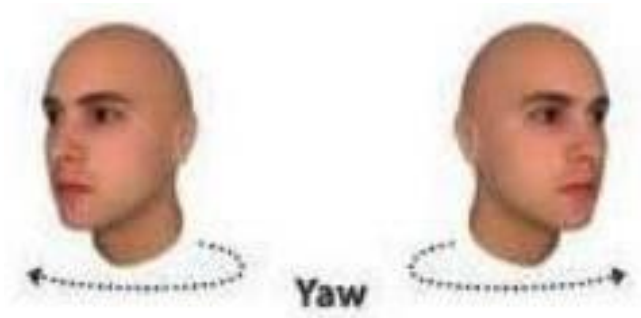
录入方向	偏转角度
正脸	正对摄像头, 无偏转角度
向上	正脸向上偏转 5~55 度
向下	正脸向下偏转 5~55 度
向右	正脸向右偏转 8~60 度
向左	正脸向左偏转 8~60 度

其中, 向上和向下偏转如下图所示。



人脸录入上下示意图

向左和向右偏转如下图所示。



人脸录入左右示意图

请不要采用如下图所示的转头方式。



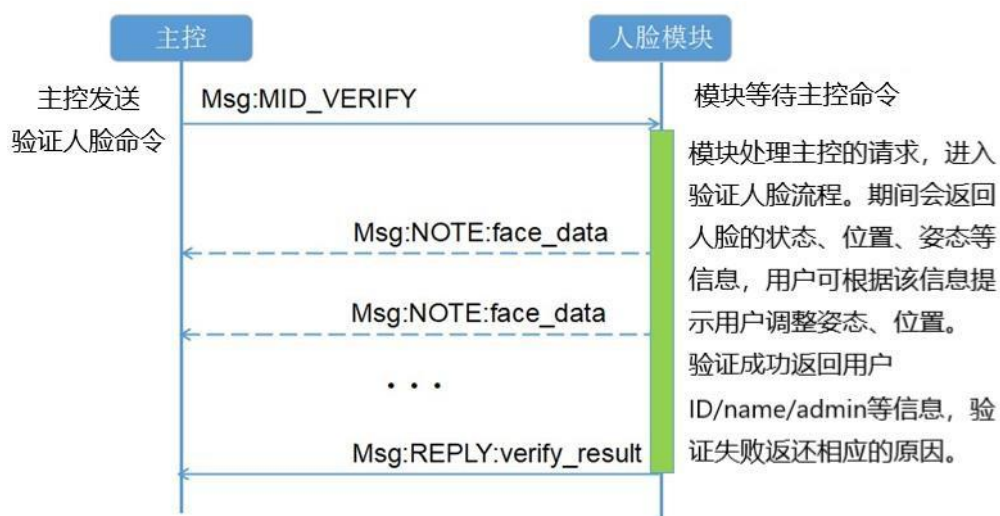
人脸录入错误示意图

我们建议用户在实际录入的过程中，缓缓向某个方向转头，并且保持偏转，请不要过快的转头或者立刻回归正脸状态。

6. 单帧录入

单帧录入相比录入流程的区别，主要在于单帧录入只需要一张正脸即可录入成功。而不需要交互五次完成注册。

7. 验证人脸流程

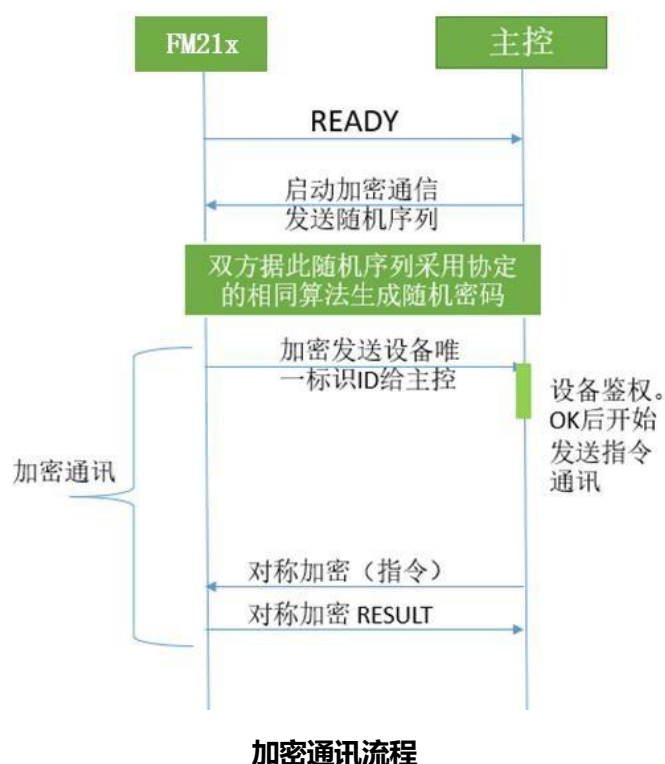


验证人脸流程

8. 加密通信流程

加密通信流程如下图所示。

- ① 主控给 FM21x 系列人脸识别模组上电
- ② FM21x 系列人脸识别模组发送READY（明文）给主控
- ③ 模组第一次开机需要调用 MID_SET_RELEASE_ENC_KEY 指令设置私有协议需要的 16bytes 序列。
- ④ 主控采用随机数生成随机序列并发送给 FM21x 系列人脸识别模组（4bytes）
- ⑤ FM21x 系列人脸识别模组和主控双方根据这随机序列，采用自定义的私有协议（生成16bytes 的密码。双方采用相同算法生成相同的密码。之后双方本次会话都采用此密码加解密。
- ⑥ FM21x 系列人脸识别模组返回状态并采用随机密码 AES/SMPL 加密发送产品序列号给主控
- ⑦ 主控解密，鉴别设备ID 确认身份后开始处理需求指令，录入或验证等。
- ⑧ 主控采用随机生成的密码，AES/SMPL 加密的方式对指令和数据加密并发送给 FM21x 系列人脸识别模组。
- ⑨ FM21x 系列人脸识别模组 用第4 步解密出来的密码解密，判断指令合法性并处理该指令。
- ⑩ FM21x 系列人脸识别模组用第4 步解密出来的密码对指令处理结果和数据加密，并发送给主控。



9. 照片下发注册示例

MID_ENROLL_WITH_PHOTO

主控(PC 端)发起照片注册, Seq 设置为 0, Photo data 为 4byte (大端模式) 照片大小, 照片类型 (普通照片 0, 加密照片 1) 为 1byte

【Seq + photo length】

模组返回 Result 和 Seq,

当 Result 为 MR_SUCCESS, 主控将 Seq 累加 1 (Seq 从 1 开始发送照片数据), Photo data 为照片数据, 照片数据分包大小为 246byte, 当最后一包不足 246byte 时, 为其实际字节数发送。

当照片发送完成后, 模组最后返回注册结果, 当 Result 为 MR_SUCCESS, 同时携带用户 ID。

示例: 发送照片大小为 2763 小照片照片, 不带用户姓名 (Send:PC 端发送给模组, Recv: 模组发送给 PC 端)

Send:ef aa f7 00 07 00 00 00 00 0a cb 01 30 《Seq=0,photolen=0x00000acb,type=1》

Recv:ef aa 00 00 06 f7 00 00 00 00 f1 《result=0,Seq=0,userid=0》

Send:ef aa f7 00 f8 00 01 5b b5 01 35 7b 8e 76 99 ed 2b fe d4 7a 9e 3c de a4 6c fe d5 84 45 3c 9c a4 65 f8 d3 7c 98 39 d7 a3 6a f9 dc 72 96 36 d3 b0 60 f2 de 70 92 25 cd b7 62 ea c8 61 81 22 c2

be 71 e2 f5 5f b0 1b ff 86 41 dd c9 67 b6 0b f6 88 5d cf e1 4f aa 23 f8 9d 50 c6 e7 47 b0 0f eb 96 92
25 d5 38 9f 35 d6 ad 61 f5 d9 63 93 31 c7 96 4c e2 f4 49 ac 0e ed 96 5f cc e7 49 ac 0e ed 96 5f cc
e7 49 ac 0e ed 96 5f cc e7 49 ac 0e ed 96 5f cc e7 49 ac 0e ed 96 5f cc e7 49 ac 0e ed 96 5f cc e7 49
ac c3 1f a4 7c f6 d5 03 9e 44 dc a5 4f fe d7 6a 9f 3f ce a5 92 3a d5 64 9e 3c de a1 6c ff d4 7a 9f 3d
df a4 6d fe d5 7b 9e 3c de a6 6e fa d0 7d 99 34 d6 ae 66 01 11 7b 2b 2c df a6 6c fd d6 79 9a 3f da
a1 69 fa d5 7b 9f 41 de a6 6e fe d1 6a 9b 2e fe 95 2c f8 c6 2a ff 92

Recv:ef aa 00 00 06 f7 00 **00 01** 00 00 f0

Send:ef aa f7 00 f8 **00 02** 3b fd d5 79 cc 54 ea 3f 34 fc e6 dc 3f c0 29 4f cc fb 97 0f 8c 57 72 94
2a c8 bc 74 e4 f0 5d b9 14 f6 8e 59 cb e3 4c a6 05 e5 e7 29 bb 93 3c d6 75 95 f7 39 ab 83 2c c6 65
85 c7 09 9b b3 1c f6 55 b5 d7 19 8b a3 0c e6 45 a5 27 e9 7b 53 fc 16 b5 55 36 fe 6a 40 ed 09 a4 46
3e cf 5d 71 de 38 9b 77 0d c7 4c 66 cf 2b 8a 68 1c d4 44 17 b8 5a f9 19 63 a5 37 1f a9 4d e8 0a 72
ba 26 0c a1 7f de 3c 40 88 18 32 93 77 d6 2e 56 9e 0a 20 8d 69 c4 26 5e 92 3a d5 64 9f 3c dc a5 6c
ff d4 7a 9f 3d de a5 6d fe d5 7b 9e 3c de a6 6e fa d0 7d 99 34 d6 ae 66 01 11 7b 2b 2d df a6 6c fc
d1 7f 9d 38 d8 a1 69 fa d5 7a 9c 4b df a5 6f fd c4 7f 9b 1d ee a2 7f bf 84 7c ff 4d cc 86 5f 7f dd 6f dc
ad 7e 15 ac f7 f6 48 cc cc ca c6 1f 2f df 6d ba 08 3e 81 9c e9 cd 94

Recv:ef aa 00 00 06 f7 00 **00 02** 00 00 f3

Send:ef aa f7 00 f8 **00 03** 62 84 1a f8 8c 44 d4 e0 4d a9 04 e6 9e 2e ba 90 3d d9 74 96 ee 3e
aa 80 2d c9 64 86 fe 0e 9a b0 1d f9 54 b6 ce 1e 8a a0 0d e9 44 a6 de ef 7d 51 fe 18 bb 57 2d e7 6c
46 ef 0b aa 48 3c f4 64 77 d8 3a 99 79 03 c5 57 7f c9 2d 88 6a 12 da 46 6c c1 5c ff 1b 61 ab 39 1d
b2 54 ee 0c 70 b8 28 02 a3 47 e6 3d 47 89 1b 33 9c 76 d5 35 56 9e 0a 20 8d 69 c4 26 5e 92 24 d5
77 9d 3d df a6 7c fd c4 7b a1 3c 2b ff 02 0b 9e 8f 37 63 7e 0c 40 a5 e4 fa cd 03 95 50 a2 da a9 26
b4 5d 01 05 e6 5b 9e e7 cc 3c a1 31 38 14 bc 1e c6 df f3 48 6d 52 6e f0 6d ae ce 95 e3 13 2b 6e ad
95 c7 8b 14 6b cf 29 01 fe 54 e9 bc 54 4e 84 9e 90 23 e3 9e 54 e7 a5 d0 e3 2d 74 ed 83 ed d1 2c 60
94 3a 33 b2 2e 4a 30 75 bc 8d b1 14 5b 69 e8 e8 37 46 92 fe 83 52 51 cb b4 32 36 13

Recv:ef aa 00 00 06 f7 00 **00 03** 00 00 f2

Send:ef aa f7 00 f8 **00 04** 25 4d 60 7a 84 a2 1a e1 25 6a e6 c4 06 3b 56 a9 9d cb 27 ec 2e ee
85 3f 3e d9 88 0b 38 a4 53 c0 97 b7 d2 55 7f e5 3c 07 6f 01 a4 d0 a8 2c 53 20 bb d9 f2 b2 5d 3c c4
a7 84 c4 29 01 86 b4 04 8b d5 6a 01 14 99 a1 51 64 15 99 19 10 88 8f 9e bc be c7 a1 81 ac 8a 36 15
54 e3 fc 7b 97 59 34 e4 9d f1 0a cb 92 aa a4 ff 65 ae 0e 8c 73 23 e7 df 83 aa b9 3c 83 64 cf 26 a2 12
f8 ad a1 6e 27 86 14 a8 fe d9 5f b3 84 e5 8f 42 47 2d 22 7b 3a 8b 4e 78 2c 0f 65 b1 d0 0c 16 26 c0
23 b7 a9 92 62 8b fb 92 3a 5b a9 6c a4 13 39 46 9b 11 29 3e db 1d 64 45 4b 5a c6 18 d3 62 ae 2f c9
4b 84 65 a2 18 be fd d1 a7 89 de 3c a2 6f 58 8c 54 9d 2a 6a c3 3f c0 5f aa 1e 9b e8 a5 a3 a7 5a b3
85 cf 9e a6 0d a3 36 ea 75 f0 9d d4 6e a1 34 ce d9 8e 9b 8c 98 2c 07 8c 6b 4e

Recv:ef aa 00 00 06 f7 00 **00 04** 00 00 f5

Send:ef aa f7 00 f8 **00 05** 80 4b b9 d8 bf 28 6f a6 34 f1 7d 22 ab 6a 53 2f 9a 98 2e 8f 26 59 45
b0 42 2b b1 76 52 74 9b 0f 73 47 2c 26 29 45 e3 44 ad a8 38 a4 74 d7 bb 75 5e 90 b9 8a b7 c9 41
c5 12 bc 62 cd f1 b7 92 bb d0 a2 20 74 46 f4 7b e3 70 4a 53 a3 d8 1e 4c 34 e1 48 ea 39 78 f2 e7 7a
fb b0 6f 8c 4d 11 f4 72 94 13 a2 07 1f 3c b0 4e 3f e0 b2 46 f9 2f 14 8f ac 85 84 47 44 af c5 89 20 e6
1b 35 75 b4 37 da 8b 5e 62 c8 bd c8 d3 04 80 12 df 82 54 4e 9f a4 73 f2 a3 88 a7 6b 0c 25 8f 78 49
c6 bd e6 94 3a 29 f4 86 f5 29 ab f8 ce a6 33 8a d2 a8 ab 38 68 54 c5 da 1f fa e3 75 1a d7 70 a9 fe
b2 26 09 99 d3 f3 63 3c f6 e6 af c3 0f 6f c3 7c 90 b6 6c d2 8f 78 8e e4 c9 10 89 91 47 58 98 be cf cf
e8 ac fd ff a0 11 23 11 0d a2 16 04 5a f3 b3 06 dc 45 c7 da 36 16 ca 82 f9

Recv:ef aa 00 00 06 f7 00 **00 05** 00 00 f4

Send:ef aa f7 00 f8 **00 06** ee e3 2d f1 78 ca e6 ae a1 17 7c 8a 74 7d 93 56 46 7c 07 9d 03 14 c2
4e 91 86 2a 34 ea 13 09 e7 c5 96 fa 12 fe ff 96 fc 44 92 2d bf 10 d4 44 e8 d9 58 77 88 ad 8e 1f 7d
91 a5 74 1a b9 58 56 ba ec b9 2e 21 2f 81 36 54 4a 12 22 78 52 58 ef 8b bb dc c5 86 9c cb 18 56 d4
f5 c1 08 55 c1 68 f0 f4 6a 06 e2 64 a2 46 03 8d 20 a4 10 e4 70 99 f7 92 06 d2 b6 c6 97 01 00 c9 7e
00 6c 08 bc a7 ec 91 f6 65 84 00 af ad c0 89 77 e3 06 ea 46 b1 32 fd 98 ae c6 6c be fd 3a a9 10 f2 f9
a3 44 bb 40 d4 85 08 87 33 c0 88 bf 1b fb f2 4f 97 a2 8d 6d de 2d c2 01 89 09 36 f7 61 27 3b 3e 86
bb dc 14 f5 bf ec fa 8c 79 82 43 1c 30 24 5c 2d 78 08 6e 44 79 5f 3d 89 08 7f f3 8d 94 92 a1 70 28
c8 01 d5 87 f9 de 5f 90 56 50 87 93 1c 9f 78 e2 e9 af e2 69 a1 bc 26 9c 0c

Recv:ef aa 00 00 06 f7 00 **00 06** 00 00 f7

Send:ef aa f7 00 f8 **00 07** d5 e1 75 5d ff f0 6d 38 51 10 3d 1e bc c0 17 38 c7 26 b3 a5 6a 81 2c
1d b6 4a 53 9c 82 a1 60 be ec 24 ee f0 8a b2 cf ea a8 b8 89 70 27 36 5a 42 77 12 f6 43 45 2b b3 a6
8c c0 16 63 17 fa 9f 55 7e 7d 55 d2 28 9c 61 9f 0a f0 c6 cc 81 73 5b 14 d8 f4 4c 99 93 b3 4f 12 3e 73
67 a5 e0 ba db a9 c0 33 99 58 e6 56 28 9f 10 c9 57 f3 86 05 c9 12 59 98 11 c6 89 de c3 5a 79 29 91
46 b8 41 74 ad 46 b5 72 b1 42 03 c3 5b f4 5d ad c7 08 0a c3 4b 9e 46 42 5c 5e 3b 55 03 05 0d 4d
d8 7d 3b 23 88 07 4b 10 95 43 f3 b7 ea ad 31 0d 39 19 45 cb 1f f4 bc 96 b5 ca 8f 19 c1 28 ac 54 74
97 d4 61 cb 3f 0a 56 b2 80 dd 27 90 70 0b 4f e9 d6 62 b2 b6 64 50 a6 13 65 e5 a0 32 d6 c1 ce 89 9f
45 9a f5 1e 82 0d e1 63 ee a7 b7 e8 ea a1 52 54 7f b4 6f c2 4f 55 2e 43 21 0f

Recv:ef aa 00 00 06 f7 00 **00 07** 00 00 f6

Send:ef aa f7 00 f8 **00 08** bd 13 35 d7 fb 09 0c b0 bc 56 0b eb 30 d0 bc 4e 53 bd 92 47 b3 91
2d 50 ed 7a d6 f8 e3 6e cd e3 d5 1a e5 56 c8 08 60 a3 85 87 f1 b9 38 ab 52 ad 52 ed 8f 46 99 40 b5
cd 38 ac a1 ae 17 7a 3a bd b7 49 e7 78 ae e7 06 da e6 cd 7d 2c 8d 65 85 db 4e 0d d8 7b fc 7d 59 67
cd d7 43 82 6a 0a 10 0e 1b 1f 8a ca 89 bf 1f b2 e2 27 13 2e 92 ac 0d 7f 55 b8 58 18 96 0d fc 2d 31
43 f6 d7 c3 df cc 0b 53 4c 3b 0e 52 c7 53 c6 c0 97 45 81 b0 4e ea ac 8d e2 80 a4 0e e4 c7 bd f2 84
8d fc 9c 5e 64 c0 dd c3 7c d4 b4 77 03 7f 9a b2 a2 67 ea 25 11 ca 30 19 c2 9f b3 28 2a 6d 37 29 d0
31 5e f4 3b df 7a 9c d1 da 74 3e b0 d8 91 e3 48 c3 e4 bb 76 0a 63 58 12 54 17 17 e8 90 d6 77 f1 74
04 73 f4 6e cc da 27 48 c0 e9 46 25 a4 f6 a6 f5 16 01 94 be d0 92 5b 49 77 7a 83

Recv:ef aa 00 00 06 f7 00 **00 08** 00 00 f9

Send:ef aa f7 00 f8 **00 09** 7e e6 67 48 27 03 9d d1 ec 51 0d 86 2c c6 8d 0f d8 de b2 44 7d 55
48 8e 59 bd d5 f3 94 08 d7 0c 09 07 44 2c 16 7a db e4 9a 31 60 69 2d c3 e5 dc d4 91 15 18 e5 59
8d c8 d2 0a b7 f5 7b 64 e9 f7 41 6b 0f cf 18 16 e8 3b 70 9f 1a d7 4a d0 ba 06 92 7d 1e c5 b8 0e 57
fc df f4 0d 2d 90 d5 c2 0e 92 44 c0 f3 7c 86 14 9f 83 87 b8 ff ee 40 84 50 1f 1e 8e ec 5f 04 67 00 ab
b7 07 59 b9 ac fc 43 28 9e 74 13 06 85 55 a3 a2 08 a0 0c df cb ad e2 2b e5 8a 18 31 c9 8b 59 44 59
7d ec 60 d6 23 5e 62 60 94 8d 27 87 1c db 43 4e 74 15 f9 5f 29 1e a4 4d 30 a5 d8 8b 8f b5 98 d8 3e
3c 37 d3 1f 9e 00 8f 5e b3 94 d6 67 eb 72 51 ad 91 49 78 ac 7f 69 4e 6d 26 16 5e 22 06 09 3a 14 fe
44 d9 73 4d fd 5c 22 fd 85 21 ee 6b bf c0 84 81 6c 26 b3 fc b4 82 af 58 d9 14 88

Recv:ef aa 00 00 06 f7 00 **00 09** 00 00 f8

Send:ef aa f7 00 f8 **00 0a** 17 bc b5 db ac 60 be 47 17 21 09 6a 18 be 22 f0 59 6a d5 41 d3 53
66 25 23 72 c3 84 8e ac 4d 5d e8 2a 0f 1f 03 09 29 e8 46 d5 9e 4c d3 a0 62 30 37 62 6d 6f 91 e4 cc
fd bd 08 87 72 78 ff 40 09 e0 c4 7a cd 68 e3 2d 6d 2b 60 fe eb b7 6b 27 0d 62 a2 dc 9f 36 20 4b 4d
da 56 a1 a9 55 c5 b2 a3 16 5d 5c fa b1 53 22 9a ee 5e 15 1f ea cb 23 64 36 55 5a 8c 66 63 74 8b aa
69 f2 08 e7 72 cc d7 77 aa 88 d1 29 f8 8e a5 2e cd 5b 6d 39 fb b5 cc 24 7c 2d 6a b3 0d 5f 60 49 8e
58 37 41 b2 b9 96 78 58 0d b1 c1 40 23 8c 65 84 b9 c4 fa df 40 70 7c d6 58 07 c8 69 a8 01 11 6d 7b
a1 08 d7 d4 dd 68 4a 2e bb 20 f3 fd d8 3e 3e f7 04 00 f1 46 b5 72 f4 5c 08 c1 6f eb a6 20 bb be a1

8f ba 4a bd 58 bb ba 53 ab c1 cb 45 79 d4 a7 7f fa cb f9 0a a6 37 e0 c0 83 db be

Recv:ef aa 00 00 06 f7 00 **00 0a** 00 00 fb

Send:ef aa f7 00 f8 **00 0b** fe 81 4e 39 23 55 a4 7c 77 3e 92 05 19 08 fd 8b a6 b4 5f 28 cf a6 61
ba 31 36 12 b8 c0 0c c7 fd d0 2d 57 90 2c ae 9f cd 47 88 c5 57 5e 92 fe 20 d1 f8 95 c0 25 85 d4 71
0c 08 ef eb 25 4b a8 27 b3 c9 e7 11 ab 55 e1 74 d4 d0 d6 ce 70 9f a3 3f 30 d3 e5 f7 42 e7 b7 30 f6
fe f6 c8 6c 54 d5 62 a0 a3 47 0e fd 51 a3 f0 c7 13 df 5d 08 b8 2a 98 63 0a 18 bf ae 24 f8 9d 6f cf 67
43 07 fb e8 44 cd b4 ae a2 a4 c1 43 9b 54 b8 7f 51 28 5e bb c0 b2 17 47 c2 3c 28 45 ec a2 51 f3 53
d6 20 d5 91 61 b0 14 7d 1c 37 c3 fd c9 89 ee 7c 38 0b b2 a8 4a ca e3 9a 13 3b da f5 8d 5b 28 8d 8e
81 9e 66 23 60 d3 c0 21 4e 9f da 44 03 28 8f 85 ef 81 4e f1 73 88 04 c0 db 4d 95 fc ed 0c d3 69 87
b6 77 f1 73 fe 7a d2 14 06 03 97 0f 56 9d ec c3 a6 71 a1 f8 9b a9 01 f9 e5

Recv:ef aa 00 00 06 f7 00 **00 0b** 00 00 fa

Send:ef aa f7 00 3b **00 0c** 77 02 96 95 8f fb 54 aa e2 db 08 57 9e 55 73 79 51 98 e7 0a 81 05
15 ff 23 e6 05 8d d9 b6 e6 14 76 80 72 22 ae 91 d5 e1 6c aa 45 0a 6d 9a dd 43 64 6c 91 34 ee a5 3b
92 27 24

Recv:ef aa 00 00 06 f7 00 **00 0c 00 01** fc 《发送完成， UserID=1》

(三) 补充说明

1. MID_REPLY

主控发送给模组的每条命令，最终都会收到模组的 MID_REPLY 回复，该消息包含主控发送的命令 mid，命令的执行结果 result，以及可能返回的数据 data。

```
typedef struct {  
    uint8_t mid; // the command(message) id to reply (the request message ID)  
    uint8_t result; // command handling result: success or failed -> s_msg_result  
    uint8_t data[0];  
} s_msg_reply_data;
```

命令的执行结果 result 可能是以下几种情况：

```
/* message result code */  
typedef uint8_t s_msg_result;  
const uint8_t MR_SUCCESS = 0; // success  
const uint8_t MR_REJECTED = 1; // module rejected this command  
const uint8_t MR_ABORTED = 2; // algo aborted  
const uint8_t MR_FAILED4_CAMERA = 4; // camera open failed  
const uint8_t MR_FAILED4_UNKNOWNREASON = 5; // UNKNOWN_ERROR
```

```

const uint8_t MR_FAILED4_INVALIDPARAM = 6; // invalid param
const uint8_t MR_FAILED4_NOMEMORY = 7; // no enough memory
const uint8_t MR_FAILED4_UNKNOWNUSER = 8; // exceed limitation
const uint8_t MR_FAILED4_MAXUSER = 9; // exceed maximum user number
const uint8_t MR_FAILED4_FACEENROLLED = 10; // this face has been enrolled
const uint8_t MR_FAILED4_LIVENESSCHECK = 12; // liveness check failed
const uint8_t MR_FAILED4_TIMEOUT = 13; // exceed the time limit
const uint8_t MR_FAILED4_AUTHORIZATION = 14; // authorization failed
const uint8_t MR_FAILED4_READ_FILE = 19; // read file failed
const uint8_t MR_FAILED4_WRITE_FILE = 20; // write file failed
const uint8_t MR_FAILED4_NO_ENCRYPT = 21; // encrypt must be set
const uint8_t MR_FAILED4_NO_RGBIMAGE = 23; // rgb image is not ready
/* message result code end */

```

返回的 data 根据 mid 而有所不同，在头文件 message.h 中，以 “s_msg_reply_” 开头的结构体都是各个指令发送 reply 消息时所携带的 data，以 verify 为例如下：

```

/* message reply data definitions */

typedef struct {

uint8_t user_id_heb;

uint8_t user_id_leb;

uint8_t user_name[USER_NAME_SIZE];

uint8_t admin;

uint8_t unlockStatus;

} s_msg_reply_verify_data;

```

2. MID_NOTE

MSG::NOTE 为模组向主控主动上报的信息，例如当模组上电后会主动向主控发送一条 NOTE，表明自己已经 READY，可以接收主控的命令，主控收到该NOTE 消息后即可开始发送录入或者验证命令。

NOTE 消息中所包含的数据为：

```

typedef struct {

uint8_t nid; // note id

```

```
uint8_t data[0];
} s_msg_note_data;
/* module -> host note end */
```

3. MID_RESET

主控发送这条命令后，模组将取消之前正在执行的命令（例如录入、解锁等），返回STANDBY 状态。

4. MID_GETSTATUS

主控发送该命令后，模组返回自身当前的状态，主要包括：MS_STANDBY(0):模组处于空闲状态，等待主控命令 MS_BUSY(1):模组处于工作状态 MS_ERROR(2):模组出错，不能正常工作 MS_INVALID(3):模组未进行初始化

5. MID_VERIFY

MSG::VERIFY 为最常用功能，其携带的 data 如下：

```
// verify
typedef struct {
uint8_t pd_rightaway; // power down right away after verifying
uint8_t timeout; // timeout, unit second, default 10s
} s_msg_verify_data;
```

[pd_rightaway]表示是否在解锁后立刻关机；[timeout]为解锁超时时间（单位 s），默认为 10s，可以在发送解锁命令时设置，超时时间用户可以任意设定，最大不超过 255s。

解锁过程中，模组返回 NOTE 和REPLY 两种消息。

NOTE 主要返回的是当前帧图片中人脸的状态，以及人脸位置和姿态。

REPLY 主要返回的是解锁过程执行完毕后的最终结果，可能返回的消息包括：MR_SUCCESS, MR_FAILED4_NOSUCHFACE, MR_ABORTED 等，当返回结果是 MR_SUCCESS 时，表示解锁成功。REPALY 会携带是什么样的方式解锁。数据结构如下：

```
typedef struct
{ uint8_t user_id_heb;
uint8_t user_id_leb;
uint8_t user_name[USER_NAME_SIZE];
uint8_t admin;
```

```
uint8_t unlockStatus;  
} s_msg_reply_verify_data;
```

6. MID_ENROLL/MID_ENROLL_SINGLE

MSG::ENROLL 也是比较常用的功能, 其附带的 data 如下:

```
// enroll user  
typedef struct {  
    uint8_t admin; // the user will be set to admin  
    uint8_t user_name[32];  
    s_face_dir face_direction;  
    uint8_t timeout;  
} s_msg_enroll_data;
```

[admin]指明该录入的人为管理员; [timeout]为录入过程中的超时时间 (单位 s), 默认为 10s, 可以在发送录入命令时设置, 超时时间用户可以随意设置, 最大不超过 255s; [face_direction]仅 在 MID_ENROLL中有效, 表示用户要录入的人脸方向, 有效的人脸方向有 5 种, 定义如下:

```
/* msg face direction */  
typedef uint8_t s_face_dir;  
const uint8_t FACE_DIRECTION_UP = 0x10; // face up  
const uint8_t FACE_DIRECTION_DOWN = 0x08; // face down  
const uint8_t FACE_DIRECTION_LEFT = 0x04; // face left  
const uint8_t FACE_DIRECTION_RIGHT = 0x02; // face right  
const uint8_t FACE_DIRECTION_MIDDLE = 0x01; // face middle  
const uint8_t FACE_DIRECTION_UNDEFINE = 0x00; // face undefine
```

录入过程中同样也会返回 NOTE 和REPLY 两种消息。

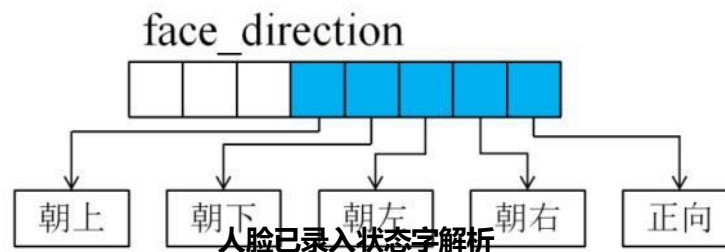
NOTE 主要返回的是当前帧图片中人脸的状态, 以及人脸位置和姿态。

REPLY 主要返回录入过程执行完毕后的最终结果, 当返回 MR_SUCCESSS 时, 表示录入成功。REPLY 同时也会携带一些信息返回, 如下所示:

```
typedef struct  
{  
    uint8_t user_id_heb;  
  
    uint8_t user_id_leb;  
  
    uint8_t face_direction;
```

```
} s_msg_reply_enroll_data;
```

其中[face_direction], 数据的低 5 位从高到低分别表示人脸方向的朝上、朝下、朝左、朝右和正向, 如下图所示, 1 表示该朝向已录入, 0 表示该朝向未录入。



7. MID_ENROLL_ITG

MSG::ENROLL_IT 是对1.15.6 ENROLL 命令的补充和扩展, 录入类型可以在参数结构中指定, 并可以让用户选择是否重复录入;

其附带的 data 如下:

```
// enroll user intergrated
```

```
typedef struct {
```

```
uint8_t admin; // the user will be set to admin, 1:admin user, 0:normal user
```

```
uint8_t user_name[USER_NAME_SIZE];
```

```
uint8_t face_direction; // reference FACE_DIRECTION_*
```

```
uint8_t enroll_type; // reference FACE_ENROLL_TYPE_*
```

```
uint8_t enable_duplicate; // enable user enroll duplicatly, 1:enable, 0:disable
```

```
    // when enroll_type is equal to FACE_ENROLL_TYPE_RGB, the enable_duplicate can be set  
    to 2, it means that cant duplicate enroll with username
```

```
uint8_t timeout; // timeout unit second default 10s
```

```
uint8_t reserved[3]; // reserved feild
```

```
} s_msg_enroll_itg;
```

[enroll_type] FACE_ENROLL_TYPE_INTERACTIVE 时, 为交互录入; FACE_ENROLL_TYPE_SINGLE 为单帧录入

[enable_duplicate]功能如下:

- 0: 不可以同一个人录入, 但是 user name 可以重复。
- 1: 可以同一个人重复录入, user name 也可以重复。
- 2: 可以同一个人录入, 但是 user name 不可以重复(仅在RGB 注册时有效)。

8. MID_DELUSER

MSG:DELUSER 为删除一个已注册的用户, 通过 user_id 指定。

9. MID_DELALL

MSG::DELALL 为删除所有注册的用户信息。

10.MID_GETUSERINFO

主控通过 user id 指定要返回的注册用户, 模组接收到该命令后会返回指定用户的信息, 主要包括以下信息:

```
typedef struct
{
    uint8_t user_id_heb;
    uint8_t user_id_leb;
    uint8_t user_name[32];
    uint8_t admin;
} s_msg_reply_getuserinfo_data;
```

11.MID_ENROLL_WITH_PHOTO

主控发起照片注册, Seq 设置为 0, Photo data 为 4byte (大端模式) 照片大小;

模组返回 Result 和 Seq,

当 Result 为 MR_SUCCESS, 主控将 Seq 累加 1 (Seq 从 1 开始发送照片数据), Photo data 为照片数据, 照片数据分包大小为 246byte, 当最后一包不足时, 为其实际字节数发送。

12.MID_DEMOMODE

控发送该命令后，模组进入演示模式，这种模式下模组鉴权的流程中将不做特征比对，即所有人都可以解锁，但是仍然会进行活体检测。

13.MID_GET_ALL_USERID

获取所有已注册用户的数量和所有已注册用户的 ID。