

JMY680S 使用手册

Revision 1.1

1 概述

JMY-680 系列射频读写模块采用 PHILIPS 公司的非接触技术设计的微型嵌入式非接触式 IC 卡读写模块。内嵌 MF RC500 或 MF RC531 射频基站，用户不必关心射频基站复杂的控制方法，只需要简单地通过选定的 UART 或 IIC 接口发送命令就可以对卡片进行完全的操作。

JMY-680 具有独特的自动寻卡功能，用户不必频繁寻卡，只需要判断 ICC 引脚的电平即可得知天线区域内是否有卡片。

模块内置 512 字节的 EEPROM，用户可以存储应用数据。

2 模块型号

JMY-680M	Mifare One 专用，默认自动寻卡
JMY-680S	Mifare One + SAM，默认自动寻卡
JMY-680P	Mifare One + Mifare Pro + Ultra Light + SAM 卡，可设定自动寻卡
JMY-680B	Mifare One + Mifare Pro + Ultra Light + TYPE-B + SAM 卡，可设定自动寻卡
JMY-680W	TYPE-A + TYPE-B Wiegand 格式输出（包含 16 种以上非接触卡）

当前可供货型号：

JMY-680MI	IIC 接口
JMY-680MST	UART 接口，TTL 电平
JMY-680MSR	UART 接口，RS232 电平
JMY-680SI	IIC 接口
JMY-680SST	UART 接口，TTL 电平
JMY-680SSR	UART 接口，RS232 电平
JMY-680PI	IIC 接口
JMY-680PST	UART 接口，TTL 电平
JMY-680PSR	UART 接口，RS232 电平
JMY-680BI	IIC 接口
JMY-680BST	UART 接口，TTL 电平
JMY-680BSR	UART 接口，RS232 电平
JMY-680W	Wiegand 34 接口

3 接口通讯协议

3.1 UART 协议

- 异步半双工，1 位起始位+8 位数据位+1 位停止位
- 波特率：19200
- 发送数据格式
长度字 + 命令字 + 数据域 + 校验字
- 长度字：指明从长度字到数据域最后一字节的字节数
- 命令字：本条命令的含义
- 数据域：此项可以为空
- 校验字：从长度字到数据域最后一字节的逐字节异或值

3.2 返回数据格式

- 成功：长度字 + 接收到的命令字 + 数据域 + 校验字
- 失败：长度字 + 接收到的命令字取反 + 校验字

3.3 IIC 协议

- 模块 IIC 地址为 0xA0
- IIC 通讯速率：400K
- 数据格式：
(模块地址+W/R) + 长度字 + 命令字 + 数据域 + 校验字
例如：模块地址为：0xA0，写则 bit0 为 0，则读指令为：0xA0 + 0x00 = 0xA0
- 长度字：指明从长度字到数据域最后一字节的字节数
- 命令字：本条命令的含义
- 数据域：此项可以为空
- 校验字：从长度字到数据域最后一字节的逐字节异或值

3.4 返回数据格式

- 成功：长度字 + 接收到的命令字 + 数据域 + 校验字
- 失败：长度字 + 接收到的命令字取反 + 校验字

3.5 Wiegand 格式

- 此处的 Wiegand 格式为 Wiegand 34。其中，第一个字节的前半字节代表卡的类型：
 - 0x0: Mifare One S70
 - 0x1: Mifare One S50
 - 0x2: Mifare Pro
 - 0x3: Ultra Light
 - 0x8: 标准 TYPE-B 卡
 - 0x9: AT88RF080

3.6 通讯端口定义

JMY-680 系列模块同时支持 IIC 和 UART 通讯接口,用户在订货时指定接口为 IIC 或 UART,当接口为 UART 时,还可由用户指定接口的电平类型,即:RS232 电平或 TTL 电平。

当通讯端口为 UART 时,波特率 19200bps。当通讯端口为 IIC 时,最大通讯速率 400K。

4 JMY-680S 通讯协议命令列表

	命令名称		长度字	命令字	数据及说明
1	模块控制	发送	0x03	0x11	1 字节工作控制字 天线状态 → Bit0=0: OFF Bit0=1: ON 自动寻卡 → Bit1=0: OFF Bit1=1: ON
		返回	0x02	0x11	
2	设置 LED	发送	0x03	0x13	1 字节 LED 状态 1 = 亮; 0 = 灭

		正确 返回	0x02	0x13	
		错误 返回	0x02	0xEC	
3	设置蜂鸣器	发送	0x03	0x14	1 字节蜂鸣时间, 单位 10ms
		正确 返回	—	0x14	
		错误 返回	0x02	0xEB	
4	读 EEPROM	发送	0x05	0x15	2 字节地址 (高字节在前) + 1 字节读取字节数 (单次操作最多 16 字节)
		正确 返回	—	0x15	读取的字节数目的数据
		错误 返回	0x02	0xEA	
5	写 EEPROM	发送	—	0x16	2 字节地址 (高字节在前) + 1 字节读取字节数 + 相应字节的写入数据 (单次操作最多 16 字节)
		正确 返回	0x02	0x16	
		错误 返回	0x02	0xE9	
6	寻卡	发送	0x03	0x20	1 字节寻卡模式 寻卡模式 → =0: 寻天线区内所有卡 =1: 寻未进入休眠状态 的卡
		正确 返回	—	0x20	4 或 7(Ultra Light)字节卡序列号 + 2 字节卡类型号
		错误 返回	0x02	0xDF	

7	读块	发送	0x0A	0x21	1 字节密钥标识+1 块号+6 字节密钥 密钥标识 → =0: A 密钥 =1: B 密钥 块 号 → =绝对块号 (1~255)
		正确 返回	0x12	0x21	16 字节数据
		错误 返回	0x02	0xDE	
8	写块	发送	0x1A	0x22	1 字节密钥标识+1 块号+6 字节密钥+16 字节写入数据
		正确 返回	0x02	0x22	
		错误 返回	0x02	0xDD	
9	初始化 钱包	发送	0x0E	0x23	1 字节密钥标识+1 块号+6 字节密钥+4 字节钱包 初始值 (低字节在前)
		正确 返回	0x02	0x23	
		错误 返回	0x02	0xDC	
10	读钱包	发送	0x0A	0x24	1 字节密钥标识+1 块号+6 字节密钥
		正确 返回	0x06	0x24	4 字节钱包值 (低字节在前)
		错误 返回	0x02	0xDB	
11	充值	发送	0x0E	0x25	1 字节密钥标识+1 块号+6 字节密钥+4 字节增加 值 (低字节在前)
		正确 返回	0x02	0x25	

		错误 返回	0x02	0xDA	
12	扣款	发送	0x0E	0x26	1 字节密钥标识+1 块号+6 字节密钥+4 字节扣款 值（低字节在前）
		正确 返回	0x02	0x26	
		错误 返回	0x02	0xD9	
13	备份钱包 值	发送	0x0B	0x27	1 字节密钥标识+1 当前钱包块号+1 字节备份钱 包块号+6 字节密钥
		正确 返回	0x02	0x27	
		错误 返回	0x02	0xD8	
14	卡休眠	发送	0x02	0x28	
		正确 返回	0x02	0x28	
		错误 返回	0x02	0xD7	
15	设置 SAM 卡波特率	发送	0x04	0x50	1 字节 SAM 卡座编号，0 或 1 + 1 字节波特率代码，0：9600；1：38400
		返回	0x02	0x50	
16	SAM 卡复 位	发送	0x02	0x51	
		返回	—	0x51	复位信息，长度由卡片决定
17	SAM 卡 COS 命令	发送	—	0x53	COS 命令
		返回	—	0x53	返回结果

5 命令例子

5.1 UART 命令

5.1.1 Mifare One S50/S70

0x0A210001FFFFFFFFFFFF2A 读第 1 块
0x0A2100FFFFFFFFFFFFFD4 读第 255 块 (S70)
0x1A220001FFFFFFFFFFFF1234567890ABCDEF1234567890ABCDEF39 写块
0x03200023 寻卡
0x021210 休眠

5.1.2 Mifare Pro

0x03300033	MP reset
0x07310084000004B6	取随机数
0x07310084000008BA	取 8 字节随机数

5.1.3 Ultra Light

0x03410143 读第 1~4 块
0x03410547 读第 5~8 块
0x0742058888888840 写第 5 块

5.2 IIC 命令

以下 IIC 命令都可执行，发送后会有结果返回。0xA0 为 IIC “写” 指令，最后一字节为校验字。

5.2.1 Mifare One S50/S70

0xA0 0A2100FFFFFFFFFFFFFD4 读第 255 块 (S70)
0xA0 1A220001FFFFFFFFFFFF1234567890ABCDEF1234567890ABCDEF39 写块
0xA0 03200023 寻卡
0xA0 021210 休眠

5.2.2 Mifare Pro

0xA0 023032	MP reset
0xA0 07310084000004B6	取随机数
0xA0 07310084000008BA	取 8 字节随机数

5.2.3 Ultra Light

0xA0 03410143	读第 1~4 块
0xA0 03410547	读第 5~8 块
0xA0 0742058888888840	写第 5 块