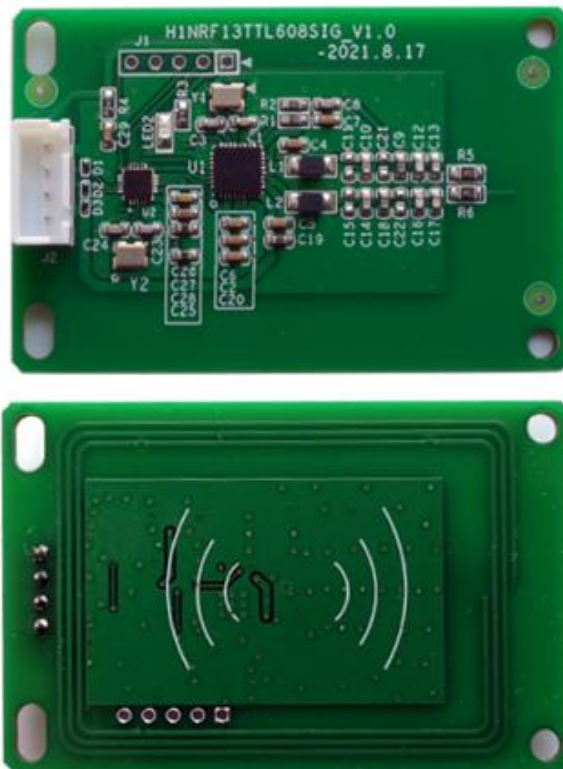


H1NRF13TTL608SIG(TTL608 射频模块)

(明月刀系列读卡模块)版本 V1.1



【模块简介】

H1NRF13TTL608SIG(明月刀系列射频读卡模块简称 TTL608)是采用兼容 RC523 非接触式读写卡芯片, 选用一线品牌物料, 支持 TTL 接口, 基于专业设备元件参数调试, 读卡性能稳定可靠, 已批量市场验证, 目前已经在以下产品或行业得到广泛应用。

- 电动自行车, 电动车或新能源充电设备;
- 物联网净水器, 水控或智能水表终端;
- 智能门锁, 门禁通道机等;
- 一卡通或共享刷卡支付设备;
- 医疗康复设备或医疗保健养生器械;
- 其他需要射频读卡应用场景;



【竞品分析】

和同类射频 IC 卡感应模块相比, 选用元器件品质好, 生产出来读卡模块读卡稳定性、一致性都优于其它厂家产品。

1、选用原装读卡芯片兼容 RC523, 产品批量读卡一致性、稳定性都得到保证, 其它厂家读卡模块有的选用引脚兼容读卡 IC, 产品故障率和读卡稳定性和原装芯片有差异。

2、选用日本 TDK 线绕电感, 电感额定电流 250mA, 读卡时发射电流瞬间增大, 电感不会饱和, 不影响读卡距离和读卡稳定性。其它厂家读卡模块用 0805 封装叠层片式电感, 物料成本上和线绕电感相差好几倍, 叠层片式电感在读卡瞬间发射电流增大, 导致电感饱和和影响读卡距离和读卡稳定性。

3、选用国内知名品牌晶振, 精度 10ppm, 温度范围-40~85℃。其它厂家读卡模块用 49S 封装晶振, 相比 3225 封装晶振成本低。另外其它厂家读卡模块晶振靠近板边, 有两方面风险, 一是, 49S 封装晶振是板上最高器件(高 5mm), 在安装或运输过程中, 受外力有碰掉风险, 对高度有要求场合不能用。二是, 27.12M 晶振奇次谐波对板上其它电路产生干扰, 有可能导致整个产品辐射或传导测试超标风险。

4、在供电电源部分做了防护, 防止因电源 EMC 或 EMI 不好影响读卡功能和距离, 谐振电容选用温度系数小 COG(NP0)材质电容, 选用国际一线品牌, 保证射频一致性和读卡的稳定性。

【技术参数】

- 工作电压: 3.3V(推荐工作电压, 工作电压低于 3.3V 会影响读卡距离)
 - 3.3V 电源纹波 $\leq 50\text{mV}$ (电源纹波大会影响读卡)
 - 可选支持 5.0V 电压版本
- 工作电流: $\leq 50\text{mA}@3.3\text{V}$
- 工作频率: 13.56MHz
- 支持 TTL 接口
- 支持 ISO/IEC14443 TypeA/TypeB 协议, ISO/IEC18092 的 P2P 模式/手机 NFC 支付应用
- 支持低功耗卡检测(LPCD)功能
- 工作温度: -30~75℃
- 储存温度: -40~85℃

【读卡距离】

读卡模块安装位置应远离金属外壳、覆铜 PCB 板等金属屏蔽物体, 以免影响读卡距离。读卡模块安装时, 天线线圈面朝外面, 方便用户刷卡。以下读卡距离是在没有外部金属板影响情况下测试, 读卡距离依 5mm 为步进, 读卡没有盲区。读卡距离也和用户卡片的频点有直接关系, 建议用户从正规厂家采购 S50、S70 卡片。

- 读卡距离 $\leq 3\text{cm}$ @ 滴胶 M1 卡(小卡)
- 读卡距离 $\leq 5\text{cm}$ @ 标准 M1 卡

【线序说明】

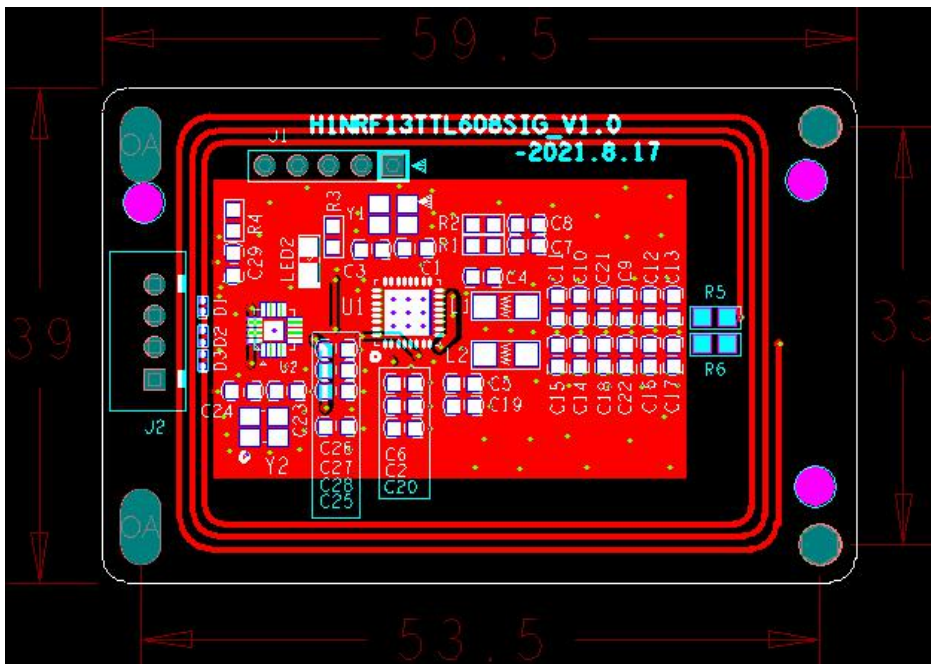
读卡模块采用 3.3V 供电，通过 TTL 接口与用户 MCU 相连。如果用户单片机是 5.0V 供电，接口需要电平转换电路，不能直接连接读卡模块。采用 XH-2.54-4AW 间距插针或者连接器，读卡模块正反面有对应丝印标注，按照上面顺序接线，和市面上其它主流厂家读卡模块线序兼容。

表 1、J2 接口说明

引脚	丝印	说明
1	VCC	电源正极，3.3V(可选支持 5.0V 电压版本)
2	URXD	串口接收(TTL)
3	UTXD	串口发送(TTL)
4	GND	电源地（串口 GND 共用此口）

【板子尺寸】

- 板子尺寸: 59.5*39mm
- 定位孔尺寸: 53.5*33mm，固定孔直径 3.2mm。



板子尺寸

【注意事项】

- 读卡模块工作时产生 13.56M 无线电干扰，可能需要用户对干扰采取切实可行措施，避免对板上其它电路产生干扰；读卡模块安装位置应远离金属外壳，大面积覆铜电路板等金属屏蔽物，以免影响读卡距离；读卡模块采用 3.3V 供电，引线过长、引线过细、单排针氧化、单排针接触不良等会导致供电电压或其它信号异常，导致不读卡或读卡不稳定。
- 读卡模块安装时，天线线圈面朝外面安装。相关驱动请联系技术支持。



读卡器串口通讯协议

V01.002

文档历史记录

版本号定义规则:

使用阿拉伯数字,并由小数点分割成两部分。

第一部分(2位有效数字):整体升级或改造时使用。

第二部分(3位有效数字):本文档重大修改时使用,通常需要修改当前生产使用的应用程序。

日期	姓名	版本	更新记录
2021-09-08	小王	V01.001	基础版本

一. 协议约定

- 1.1、其他设备(以下简称上位机)与读卡器通过串口通信。
- 1.2、约定上位机发给读卡器为下行数据、读卡器发给上位机为上行数据。
- 1.3、长度字段为从“开始符”到CRC校验,包含开始符和CRC校验,以字节为单位。
- 1.4、CRC-16校验,从“开始符”到数据体(开始符~CRC16的前1个字节)的数据。
- 1.5、多字节类型按照大端字节发送,无特殊说明,数据格式都是十六进制
- 1.6、本读卡器只支持M1卡读写操作,如果是CPU卡读写会返回卡类型错误

二. 硬件接口



2.1 采用 UART 通信

2.2 参数配置, 通信波特率: 9600bps 或者 2400bps, 通信格式 1 (起始位)、8 (数据位)、1 (停止位), 无奇偶校验.

三. 通信机制

读卡器空闲时处于等待上位机发送指令状态;

读卡器每隔 50ms 自动寻卡, 寻到卡后上报寻卡应答包

上位机对卡操作前应先进行寻卡操作, 如果当前无卡, 读卡器接收到上位机的读或写等指令时会返回无卡异常。

四 协议包格式

字段	长度	描述
开始符	2	上传: 0x24, 0x24
数据长度	2	Hex 下传数据包总长度 (开始字符 ~ CRC16 校验, 低字节在前, 高字节在后)
报文命令	1	Hex
数据体	N	
校验码	2	CRC16

命令列表:

下行报文命令	描述	上行报文命令	描述
0x10	查询读卡器版本	0x80	读卡器版本应答
0x11	寻卡	0x81	寻卡应答
0x12	读卡	0x82	读卡应答
0x13	写卡	0x83	写卡应答
0x14	钱包增值	0x84	钱包增值应答
0x15	钱包减值	0x85	钱包减值应答

五. 数据包内容

5.1 查询读卡器版本 上位机—》读卡器

报文命令	0x10
------	------



示例
24 24 00 07 10 79 0B
描述
24 24: 开始符
00 07: 数据长度
10: 查询读卡器版本命令
79 0B: 校验位

5.2 读卡器版本应答 读卡器—》上位机

报文命令	0x80		
示例	24 24 00 0A 80 01 01 01 47 B8		
数据体内容			
名称	长度	格式	描述
协议版本	1	Hex	24 24: 开始符 00 0A: 数据长度
读卡器软件版本	1	Hex	80: 读卡器应答 01: 协议版本
读卡器硬件版本	1	hex	01: 软件版本 01: 硬件版本 47 B8: 校验位

5.3 寻卡 上位机—》读卡器

报文命令	0x11		
示例	24 24 00 07 11 B8 CB		
描述			
24 24: 开始符			
00 07: 数据长度			
11: 寻卡命令			
B8 CB: 校验位			

5.4 寻卡应答 读卡器—》上位机

报文编号	0x81		
-------------	------	--	--



示例			
无卡: 24 24 00 0D 81 01 01 00 00 00 00 92 CF			
有卡: 24 24 00 0D 81 02 02 DC F0 13 05 20 6C			
数据体内容			
名称	长度	格式	描述
操作结果	1	hex	1- 无卡 2- 寻卡成功
卡类型	1	hex	1- 无卡 2- M1 卡 3- CPU 卡
物理卡号	4	HEX	无卡时数据全 0

5.5 读卡 上位机—》读卡器

报文命令	0x12		
示例			
读 1 扇区 1 块数据: 24 24 00 14 12 01 DC F0 13 05 DC F0 DC F0 13 05 01 01 70 91			
读 1 扇区 3 块数据: 24 24 00 14 12 01 DC F0 13 05 DC F0 DC F0 13 05 01 03 F1 50			
数据体内容			
名称	长度	格式	描述
扇区密钥类型	1	hex	1- KEYA 密钥 2- KEYB 密钥
扇区密钥	6	hex	
物理卡号	4	hex	请求卡时读卡器返回
扇区编号	1	hex	1K 容量的卡为 0—15 4K 容量的卡为 0—63
块号	1	hex	0-2 代表读该扇区的 0-2 块的某一块 3-代表读该扇区的密钥块, 具体数据依赖于控制字节权限控制, 如果发卡时设置不允许读, 则读取的数据为全 0 4-代表读取该扇区 0-2 块全部数据 5 代表读取该扇区全部内容, 密钥块具体数据依赖于控制字节权限控制, 如果发卡时设置不允许读, 则读取的数据为全 0



5.6 读卡应答 《读卡器—》上位机

报文命令	0x82		
示例			
1 扇区 1 块:	24 24 00 1A 82 05 01 01 20 17 11 08 20 22 12 30 20 17 11 08 00 00 88 A8 24 C2		
1 扇区 3 块:	24 24 00 1A 82 05 01 03 00 00 00 00 00 00 78 77 88 69 00 00 00 00 00 00 C0 60		
数据体内容			
名称	长度	格式	描述
操作结果	1	HEX	1- 读卡指令数据错误 2- 无卡 3- 密钥验证失败 4- 读卡异常, 正读卡时卡移走、数据块成功而密钥块失败或其他异常 5- 读卡成功
扇区编号	1	hex	1K 容量的卡为 0—15 4K 容量的卡为 0—63
块号	1	hex	0-2 代表读该扇区的 0-2 块的某一块 3-代表读该扇区的密钥块, 具体数据依赖于控制字节权限控制, 如果发卡时设置不允许读, 则读取的数据为全 0 4-代表读取该扇区 0-2 块全部数据 5 代表读取该扇区全部内容, 密钥块具体数据依赖于控制字节权限控制, 如果发卡时设置不允许读, 则读取的数据为全 0
块数据内容	N	hex	块号 0-3 时, 数据内容长度 16 字节 块号 4 时, 数据内容 48 字节 块号 5 时数据内容 64 字节

5.7 写卡 《上位机—》读卡器

报文编号	0x13		
示例			
	24 24 00 24 13 02 10 D0 01 1D F0 87 DC F0 13 05 01 01 20 21 11 08 20 23 12 30 20 17 11 08 00 00 88 A8 41 FF		
数据体内容			
名称	长度	格式	描述
扇区密钥类型	1	hex	1- KEYA 密钥 2- KEYB 密钥
扇区密钥	6	hex	



物理卡号	4	hex	请求卡时读卡器返回
扇区编号	1	hex	1K 容量的卡为 0—15 4K 容量的卡为 0—63
块号	1	hex	0-2 代表写该扇区的 0-2 块的某一块 3-代表写该扇区的密钥块 4-代表写该扇区 0-2 块全部数据 5 代表写该扇区全部内容, 密钥块具体数据依赖于控制字节权限控制, 如果发卡时设置不允许写, 会出现数据块写成功, 密钥块写失败
块数据内容	N	hex	块号 0-3 时, 数据内容长度 16 字节 块号 4 时, 数据内容 48 字节 块号 5 时数据内容 64 字节

5.8 写卡应答 《读卡器—》上位机

报文编号	0x83		
示例	24 24 00 0A 83 05 01 01 06 3D		
数据体内容			
名称	长度	格式	描述
操作结果	1	HEX	1- 写卡指令数据错误 2- 无卡 3- 密钥验证失败 4- 写卡异常, 正写卡时卡移走、数据块成功而密钥块失败或其他异常 5- 写卡成功
扇区编号	1	hex	1K 容量的卡为 0—15 4K 容量的卡为 0—63
块号	1	hex	0-2 代表写该扇区的 0-2 块的某一块 3-代表写该扇区的密钥块 4-代表写该扇区 0-2 块全部数据 5 代表写该扇区全部内容, 密钥块具体数据依赖于控制字节权限控制, 如果发卡时设置不允许写, 会出现数据块写成功, 密钥块写失败

5.9 钱包增值 《上位机—》读卡器

报文编号	0x14
数据体内容	



名称	长度	格式	描述
预留			

6.0 钱包增值应答 读卡器—》上位机

报文编号	0x84		
数据体内容			
名称	长度	格式	描述
预留			

6.1 钱包减值 上位机—》读卡器

报文编号	0x15		
数据体内容			
名称	长度	格式	描述
预留			

6.2 钱包减值应答 读卡器—》上位机

报文编号	0x85		
数据体内容			
名称	长度	格式	描述
预留			

测试卡数据

0 扇区

0 区块:DCF013053A0804006263646566676869

1 区块:000110030303060708FFFFFFFFFFFFFFF

2 区块:FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

3 区块:A0A1A2A3A4A5FF078069B0B1B2B3B4B5

1 扇区



- 0 区块:075501685F7A5D05749B0AAA0204010E
- 1 区块:201711082022123020171108000088A8
- 2 区块:0000000000000000000000F4A807005B
- 3 区块:DCF01305DCF07877886910D0011DF087

2 扇区

- 0 区块:00000000000000000000000000000000
- 1 区块:00000000FFFFFFFF0000000002FD02FD
- 2 区块:00000000FFFFFFFF0000000002FD02FD
- 3 区块:9996924EEB1118778E6944522F54B9F5

3 扇区

- 0 区块:260853300D01000001000003001E92EB
- 1 区块:260854100C01000001000003001E92EB
- 2 区块:260854110B01000001000003001E92EB
- 3 区块:276B7E09FE527F078869276B7E09FE52

4 扇区

- 0 区块:251632460E01000000000003620D3D5F
- 1 区块:251732550E01000000000003001E92EB
- 2 区块:251743170E01000000000003001E92EB
- 3 区块:276B7E09FE527F078869276B7E09FE52

5 扇区

- 0 区块:251747010E01000000000003001E92EB
- 1 区块:251747440E01000000000003001E92EB
- 2 区块:251911100000000000000002001E92EB
- 3 区块:276B7E09FE527F078869276B7E09FE52

6 扇区



0 区块:04030102E9FFFF1100000000000000FC

1 区块:FBFCFEFD160000EEFFFFFFFFFFFFFFFFF03

2 区块:00000000000000000000000000000000

3 区块:9996924EEB117F07886944522F54B9F5

7 扇区

0 区块:20210425150510000006AE005A000000

1 区块:210000005A5A5A0000000000000000E0174

2 区块:205A5A5A5A5A5A5A5A5A5A5A5A38041C

3 区块:0B4B636B1C5B18778E690F18E40D6A2A

8 扇区

0 区块:00000708000008F30000000100000000

1 区块:0B010000F4FEFFFF0B01000022DD22DD

2 区块:0B010000F4FEFFFF0B01000022DD22DD

3 区块:0B4B636B1C5B18778E690F18E40D6A2A

9 扇区

0 区块:3646446B000E0012000FFF12047D0300

1 区块:00000013642A03000051000000000000

2 区块:20180404100504000000000050264700

3 区块:FFFFFFFFFFFFFFFF078069FFFFFFFFFFFF

10 扇区

0 区块:00000000000000000000000000000000

1 区块:00000000000000000000000000000000

2 区块:00000000000000000000000000000000

3 区块:FFFFFFFFFFFFFFFF078069FFFFFFFFFFFF

11 扇区



0 区块:00000000000000000000000000000000
1 区块:00000000000000000000000000000000
2 区块:00000000000000000000000000000000
3 区块:FFFFFFFFFFFFFFFF078069FFFFFFFFFFFFFF

12 扇区

0 区块:00000000000000000000000000000000
1 区块:00000000000000000000000000000000
2 区块:00000000000000000000000000000000
3 区块:FFFFFFFFFFFFFFFF078069FFFFFFFFFFFFFF

13 扇区

0 区块:00000000000000000000000000000000
1 区块:00000000000000000000000000000000
2 区块:00000000000000000000000000000000
3 区块:FFFFFFFFFFFFFFFF078069FFFFFFFFFFFFFF

14 扇区

0 区块:01FE0000000000000000000000000000
1 区块:00000000000000000000000000000000
2 区块:00000000000000000000000000000000
3 区块:FFFFFFFFFFFFFFFF078069FFFFFFFFFFFFFF

15 扇区

0 区块:29949355CA4E88993209DA3D37F69159
1 区块:00000000000000000000000000000000
2 区块:00000000000000000000000000000000
3 区块:FFFFFFFFFFFFFFFF078069FFFFFFFFFFFFFF



测试数据

1. 查询读卡器版本

发送: 24 24 00 07 10 79 0B

接收: 24 24 00 0A 80 01 01 01 47 B8

2. 寻卡

发送: 24 24 00 07 11 B8 CB

接收: 24 24 00 0D 81 01 01 00 00 00 00 92 CF 无卡

发送: 24 24 00 07 11 B8 CB

接收: 24 24 00 0D 81 02 02 DC F0 13 05 20 6C 有 M1 卡, 卡号 DC F0 13 05

24 24 开始符

00 0D 长度

81 寻卡命令

02 寻卡成功

02 M1 卡

DC F0 13 05 物理卡号

20 6C

发送: 24 24 00 07 11 B8 CB

接收: 24 24 00 0D 81 02 03 1D 13 54 51 E3 A9 有 CPU 卡, 卡号 1D 13 54 51

3. 读卡

发送: 24 24 00 14 12 01 DC F0 13 05 DC F0 DC F0 13 05 01 01 70 91 读 1 扇区 1 块数据

24 24

00 14

12

01 KEYA 秘钥

DC F0 13 05 DC F0 扇区秘钥

DC F0 13 05 物理卡号

01 扇区编号

01 块号

70 91 crc16 校验

接收: 24 24 00 1A 82 05 01 01 20 17 11 08 20 22 12 30 20 17 11 08 00 00 88 A8 24 C2

发送: 24 24 00 14 12 01 DC F0 13 05 DC F0 DC F0 13 05 01 03 F1 50

KEYA 秘钥读 1 扇区 3 块秘钥块

接收: 24 24 00 1A 82 05 01 03 00 00 00 00 00 00 78 77 88 69 00 00 00 00 00 00 C0 60

控制字不允许读, 读取秘钥数据全 0

发送: 24 24 00 14 12 01 DC F0 13 05 DC F0 DC F0 13 05 01 04 B0 92

KEYA 秘钥读 1 扇区 012 全部 3 块数据

接收: 24 24 00 3A 82 05 01 04 07 55 01 68 5F 7A 5D 05 74 9B 0A AA 02 04 01 0E 20 21 11 08 20 23 12 30 20 17 11 08 00 00 88 A8 00 00 00 00 00 00 00 00 00 00 00 00 F4 A8 07 00 5B 4F EB

4. 写卡

发送: 24 24 00 24 13 02 10 D0 01 1D F0 87 DC F0 13 05 01 01 20 21 11 08 20 23 12 30 20 17 11 08 00 00 88 A8 41 FF

24 24

00 24



13

02 KEYB 秘钥

10 D0 01 1D F0 87 秘钥值

DC F0 13 05

01

01

20 21 11 08 20 23 12 30 20 17 11 08 00 00 88 A8 写入数据

41 FF

接收: 24 24 00 0A 83 05 01 01 06 3D

再次读 1 扇区 1 块

发送: 24 24 00 14 12 01 DC F0 13 05 DC F0 DC F0 13 05 01 01 70 91

接收: 24 24 00 1A 82 05 01 01 20 21 11 08 20 23 12 30 20 17 11 08 00 00 88 A8 96 28

新读到的数据与写入数据一致

crc16 算法

```
const uint16_t crc16_table[256]=
```

```
{  
    0x0000,0xC0C1,0xC181,0x0140,0xC301,0x03C0,0x0280,0xC241,  
    0xC601,0x06C0,0x0780,0xC741,0x0500,0xC5C1,0xC481,0x0440,  
    0xCC01,0x0CC0,0x0D80,0xCD41,0x0F00,0xCFC1,0xCE81,0x0E40,  
    0x0A00,0xCAC1,0xCB81,0x0B40,0xC901,0x09C0,0x0880,0xC841,  
    0xD801,0x18C0,0x1980,0xD941,0x1B00,0xDBC1,0xDA81,0x1A40,  
    0x1E00,0xDEC1,0xDF81,0x1F40,0xDD01,0x1DC0,0x1C80,0xDC41,  
    0x1400,0xD4C1,0xD581,0x1540,0xD701,0x17C0,0x1680,0xD641,  
    0xD201,0x12C0,0x1380,0xD341,0x1100,0xD1C1,0xD081,0x1040,  
    0xF001,0x30C0,0x3180,0xF141,0x3300,0xF3C1,0xF281,0x3240,  
    0x3600,0xF6C1,0xF781,0x3740,0xF501,0x35C0,0x3480,0xF441,  
    0x3C00,0xFCC1,0xFD81,0x3D40,0xFF01,0x3FC0,0x3E80,0xFE41,  
    0xFA01,0x3AC0,0x3B80,0xFB41,0x3900,0xF9C1,0xF881,0x3840,  
    0x2800,0xE8C1,0xE981,0x2940,0xEB01,0x2BC0,0x2A80,0xEA41,  
    0xEE01,0x2EC0,0x2F80,0xEF41,0x2D00,0xEDC1,0xEC81,0x2C40,  
    0xE401,0x24C0,0x2580,0xE541,0x2700,0xE7C1,0xE681,0x2640,  
    0x2200,0xE2C1,0xE381,0x2340,0xE101,0x21C0,0x2080,0xE041,  
    0xA001,0x60C0,0x6180,0xA141,0x6300,0xA3C1,0xA281,0x6240,  
    0x6600,0xA6C1,0xA781,0x6740,0xA501,0x65C0,0x6480,0xA441,  
    0x6C00,0xACC1,0xAD81,0x6D40,0xAF01,0x6FC0,0x6E80,0xAE41,  
    0xAA01,0x6AC0,0x6B80,0xAB41,0x6900,0xA9C1,0xA881,0x6840,  
    0x7800,0xB8C1,0xB981,0x7940,0xBB01,0x7BC0,0x7A80,0xBA41,  
    0xBE01,0x7EC0,0x7F80,0xBF41,0x7D00,0xBDC1,0xBC81,0x7C40,  
    0xB401,0x74C0,0x7580,0xB541,0x7700,0xB7C1,0xB681,0x7640,  
    0x7200,0xB2C1,0xB381,0x7340,0xB101,0x71C0,0x7080,0xB041,  
    0x5000,0x90C1,0x9181,0x5140,0x9301,0x53C0,0x5280,0x9241,  
    0x9601,0x56C0,0x5780,0x9741,0x5500,0x95C1,0x9481,0x5440,  
    0x9C01,0x5CC0,0x5D80,0x9D41,0x5F00,0x9FC1,0x9E81,0x5E40,  
    0x5A00,0x9AC1,0x9B81,0x5B40,0x9901,0x99C0,0x5880,0x9841,  
    0x8801,0x48C0,0x4980,0x8941,0x4B00,0x8BC1,0x8A81,0x4A40,
```



```
0x4E00,0x8EC1,0x8F81,0x4F40,0x8D01,0x4DC0,0x4C80,0x8C41,  
0x4400,0x84C1,0x8581,0x4540,0x8701,0x47C0,0x4680,0x8641,  
0x8201,0x42C0,0x4380,0x8341,0x4100,0x81C1,0x8081,0x4040
```

```
};  
/*****
```

函数名称: crc16
函数功能: 16 位 crc 校验函数(查表实现)
校验多项式: $X^{16} + X^{15} + X^2 + 1$

输入:
dt: 数据首地址
len: 数据长度

输出:
res: crc 校验结果
若输入数据为全 0,则返回 0xFFFF

```
*****/
```

```
uint16_t crc16(uint8_t *dt, uint16_t len)  
{  
    uint16_t i;  
    uint16_t res;  
  
    for (i=0; i<len; i++)  
    {  
        if (dt[i]!=0)  
        {  
  
            res=0;  
            for (i=0; i<len; i++)  
            {  
  
                res=(crc16_table[dt[i]^(res&0xFF)]^(res/0x100));  
            }  
            res = (res>>8) |(res<<8); //交换 crc16 校验码的高低字节  
  
            return (res);  
        }  
    }  
    return (0xFFFF);  
}
```

使用说明

- 1.测试卡数据为我们测试的时候使用的一张 M1 卡的内部数据
- 2.测试数据为按照通讯协议测试的时候记录的数据
- 3.通讯协议为读卡器支持的功能接口协议, 目前支持 4 项功能:
---获取读卡器版本(功能独立, 是否使用与卡操作无关)、---寻卡、---读卡, ---写卡
具体约定和注意细节请参考协议。

【版本更新】



郑州睿讯微电子有限公司
ZHENGZHOU MINIMUMIOT MICROMODULES CO.,LTD.

电话：0371-55611061 传真：0371-55611061

网址：www.minimumiot.com 邮箱：zhanggong@371ic.com

地址：郑州市高新区金梭路41号(高新区管委会西隔壁)西城科技大厦 A503-504 室

版本编号	更新日期	变更内容	变更人
V01.001	2021.6.15	初始版本生成	王工
V1.0	2021.8.17	校准修改部分技术参数	刘工
V1.1	2021.9.29	产品发布	张工

【购买售后】

技术支持：13686864447 张弓

官方网址：www.minimumiot.com

官方淘宝：<https://shop108277945.taobao.com/>



【公司简介】

郑州睿讯微电子有限公司成立于2014年3月，位于郑州高新技术产业开发区，创业团队汇聚有十几年工作经验的技术研发和市场营销人员，志趣相投并励志向合作伙伴提供差异化的传感器模组方案物联网应用服务。早在公司成立前的一段时间，主创人员去参加一个物联网应用峰会，在会议上受到了国外品牌的调侃和不公平待遇，为此更加坚定了核心团体挑战卓越追求理想的勇气和信心，决心全心致力于提供品质持续改善，成本不断优化，并确保如期交付的模组化产品：如**GPS/北斗授时/定位模组**，**4G(CAT1)/NB-IoT DTU**，**LoRa/RFID/串口转以太网透传等通讯模组**；以及环境运动/生物识别等MEMS传感器模组；以期望通过模组化的组件降低产品开发难度，有效缩短终端产品上市周期。产品广泛应用于：智能仪表与传感器系统，移动及共享支付、车载运输与冷链、智慧能源管理、智能安防与消防、设备远程监控与维护，资产追踪与监控、工业应用与智能制造、康复设备与医疗健康，空气监测与农业环境和智能畜牧养殖系统等众多领域，用心为合作伙伴提供有价值的服务咱把她视为这一生的光荣职业，一份为之而奋斗不息的事业，秉承“至诚 知芯 志恒迈”立志成为一个受人尊敬，并可传承的专业模组服务企业！愿和合作伙伴一道，共同携手谱写美好明天。