

SL-13.56MOD-SU 射频卡读写模块使用说明

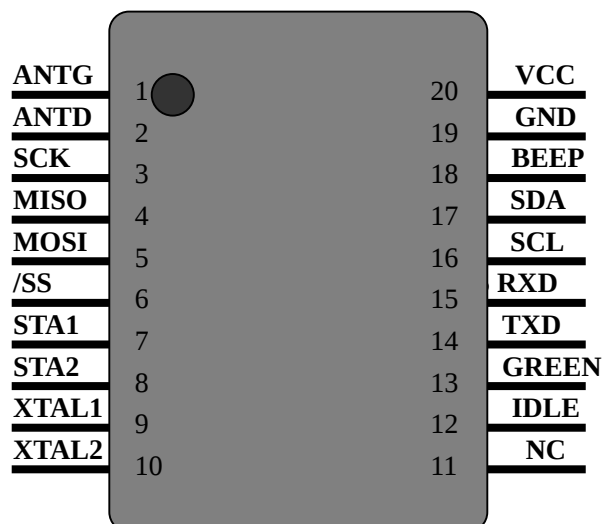
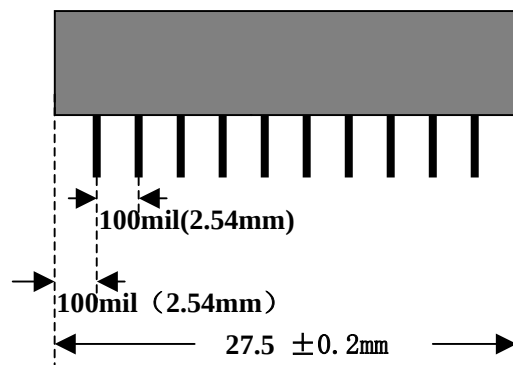
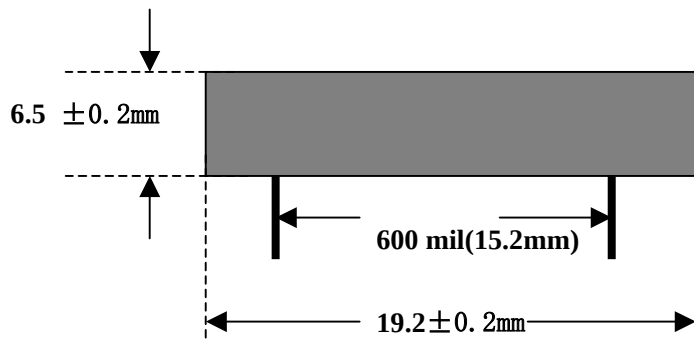
一. 概述

SL-13.56MOD-SU 射频卡读写模块，可以读写符合 ISO14443 TYPE A/B -2,-3 规范的射频卡片 (AT88RF020、AT88SC6416CRF、ST_SRI176、ST_SRIX4K、MIFARE S50、MIFARE S70 以及其他兼容 ISO14443 TYPE A/B 规范的 CPU 卡，双界面卡)。模块与上位机之间可以使用多种接口连接 (UART、SPI、TWI)，方便用户使用。

二. 特点

- 完全兼容 ISO14443 TYPE A 和 TYPE B 标准，可以稳定读写各种符合 ISO14443 A/B 两种通讯协议标准的卡片。
- 提供多种外部通讯接口，方便用户开发。
 - UART: 通用异步通讯接口。外接 RS232 芯片后可以方便的和 PC 机交换数据。
固定波特率: 115200Bps
 - SPI: 标准 SPI 硬件接口。最高速率可以达到 2.0MBps。使用标准 SPI 接口的模式 0 进行通讯。
 - TWI: 通用两线协议，可以使用最少的 I/O 数量建立模块与上位机的连接。
最高可以达到 400 KPBS。
- I/O 管脚 CMOS 和 TTL 输入/输出电平兼容
- 内置上电复位电路。
- 自带看门狗电路。
- 尺寸: 27.5mm*19mm*7.0mm。
- 目前支持卡片类型:
 1. AT88RF020
 2. AT88SC6416CRF (支持该系列的全部芯片 0104crf----6416crf)
 3. ST_SRI176
 4. ST_SRIX4K
 5. MIFARE ONE S50
 6. MIFARE ONE S70
 7. MIFARE ULTRALIGHT
 8. 兼容 ISO14443 TYPE A/B 规范的 CPU 卡，双界面卡。
- 工作频率: 13.56MHz。
- 读写距离: 读写距离和搭配天线尺寸有关。请参考天线设计的相关文档。
- 内置 256 字节 EEPROM。
- 工作电压: DC3V---5V。
- 直接驱动 50 Ohm 天线
- **卡片通讯最大 FIFO= 48 BYTE。**
- **模块最多可对外提供 13 个 I/O 口; 256 字节的 EEPROM; 支持低功耗; 带低电压检测功能。利用这些功能模块本身就可以完成一定的测控功能，可以承接应用系统的开发，将用户的业务写入模块内部，省去外部的 MCU 节省用户的硬件成本。**

三. 外观与管脚定义



管脚定义说明：

序号	名称	描述
PIN1	ANTG	天线地
PIN2	ANTD	天线驱动（50 Ohm antenna）
PIN3	SCK	SPI 接口的时钟信号
PIN4	MISO	SPI 接口的 MISO 信号线
PIN5	MOSI	SPI 接口的 MOSI 信号线
PIN6	SS	SPI 接口的片选信号（低电平有效）
PIN7	STA1	模块的工作指示（接收命令有拉底）
PIN8	STA2	模块回传数据的状态指示（数据有效后拉底）
PIN9	XTAL1	晶体输入端
PIN10	XTAL2	晶体输入端
PIN11	NC	未用
PIN12	IDLE	模块休眠（0：工作 1：休眠）
PIN13	GREEN	可以驱动一个 LED 指示灯（低有效）
PIN14	TXD	UART 接口的 TXD 信号线
PIN15	RXD	UART 接口的 RXD 信号线
PIN16	SCL	TWI 接口的时钟信号线
PIN17	SDA	TWI 接口的数据信号线
PIN18	BEEP	可以直接驱动 BEEP（低有效）
PIN19	GND	地
PIN20	VCC	电源（DC 3-5V）

四. 通讯协议

模块和上位机之间采用命令应答的通讯模式。平时模块处在空闲状态（**STA1=STA2=1**）。模块只有在空闲状态才会响应上位机的命令。模块接收到上位机的命令后，进入命令执行状态（**STA1=0,STA2=1**，在对 CPU 卡操作的过程中，在命令执行过程中，**STA1** 的状态可能会发生反转）。模块命令执行完毕，准备好执行结果后会通过状态管脚给上位机指示（**STA1=1,STA2=0**）。上位机按照指示取走命令结果后，模块再次进入空闲状态(**STA1=STA2=1**)。

对于 STA1 和 STA2 的状态关系，可以参考下表：

STA1=1	STA2=1	模块空闲，上位机可以发送命令给模块。
STA1=0	STA2=1	模块接收数据成功，开始执行上位机命令（以 STA1 的下降沿为标记，说明模块接收上位机的命令成功，开始执行命令）。模块接收到命令后最快在 100uS 以内会把 STA1 拉低，表明接收数据成功，开始执行命令。上位机如果在发送结束后 500uS 以内没有检测到 STA1 的下降沿表明，模块接收命令错误，应该重新尝试发送命令给模块。
STA1=X	STA2=1	模块执行上位机的命令（以 STA1 的下降沿为起始，上位机等待模块执行命令，这里需要做一个超时的判断。在此期间 STA1 的状态可能发生反转。如果 STA1 的状态发生反转，则需要将超时计时器清零，重新开始计时。如果在超时溢出

前都没有检测到 STA2=0 或 STA1 的反转，则表示模块工作超时。如果在超时前检测到 STA2=0 则表示模块执行命令结束。上位机可以取走执行结果)。

STA1=X STA2=0

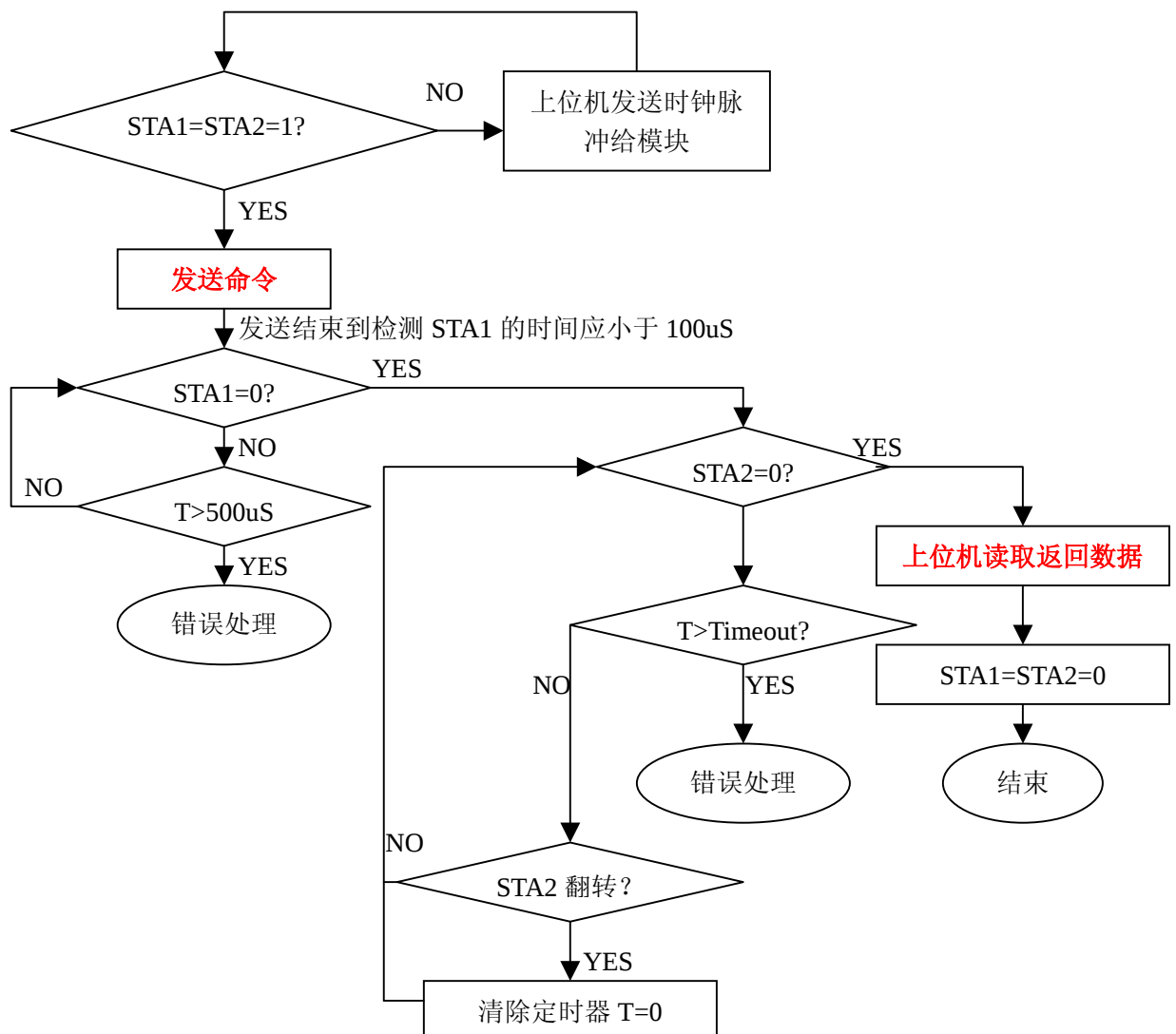
模块执行命令结束。上位机可以提取执行结果。在上位机将所有数据取走前，STA2 会保持在低电平。只有上位机将所有数据都取走后，STA2 才会恢复为高电平。对于 UART 接口，模块会主动发送命令结果给上位机，所以不会发生数据没有取完的情况发生。

STA1=1 STA2=1

模块空闲，可以发送下一条命令。

上位机可以通过 **PIN12 (IDEL)** 管脚，控制模块的休眠和唤醒。在休眠状态 (**PIN12=1**) 下，射频场关闭，MCU 休眠。此时模块功耗最小。要唤醒模块 (**PIN12=0**) 需要 >5 毫秒的延时。唤醒模块后，需要附加的开场命令打开射频场。模块复位默认射频场状态为“关闭”。

通讯中 STA1/STA2 检测流程图



其中的 **Timeout** 的取值要根据不同的卡片，有不同的要求。详见获取波特率命令。

1. 上位机发送命令给模块

数据头 + 长度 + 长度校验 + 命令代码 + 等待延时 + [命令参数] + 校验和
HEAD+LENGTH+ LEN_CHK +COMMAND+WAIT TIME+PARAMETER+CHECKSUM

- **HEAD(数据头):** 一个字节的“0xA6”。如果发送多个数据头字节,模块自动丢弃。
- **LENGTH(长度):** 整个命令数据帧中除了 HEAD(数据头)、LENGTH(长度)、LEN_CRC(长度校验)以外的所有数据的字节数。
对于没有 PARAMETER(命令参数)的命令,LENGTH=3。
对于有 PARAMETER(命令参数)的命令 LENGTH=命令参数的长度 + 3。该参数一个字节。
- **LEN_CHK(长度校验):** LEN_CHK 是 LENGTH 取反以后的值。该参数一个字节。
- **COMMAND(命令代码):** 需要模块执行的命令代码。详细的命令说明在第五部份介绍。该参数一个字节。
- **WAIT TIME(等待延时):** 该命令预计执行时间。在卡片操作命令里面这个时间是模块将读写卡命令发送完毕后等待卡片应答的时间,对于不操作卡片的命令如关场、握手等该参数可以设置为“1”;对于开场和检测卡的命令为开场的延时时间单位是毫秒;对于读写卡片的命令该参数的设置需要参考不同型号卡片的文档,根据实际需要设置。实际的等待时间是该参数的值乘以一个最小间隔单位。最小时间间隔约为“193 毫秒”。**注意该参数若设置为“0”则等待 TR0 的时间为 4969mS。**该参数一个字节。
COMMAND=10 WAIT TIME 没有意义。
COMMAND=11 WAIT TIME 单位是 19.3mS
COMMAND=12 WAIT TIME 没有意义。
COMMAND=13 WAIT TIME 单位是 1 mS
COMMAND=14 WAIT TIME 没有意义。
COMMAND=15 WAIT TIME 单位是 1 mS
COMMAND=16 WAIT TIME 没有意义。
COMMAND=17 WAIT TIME 没有意义。
COMMAND=18 WAIT TIME 单位是 19.3Ms
COMMAND=19 WAIT TIME 存储器地址。
COMMAND=1a WAIT TIME 存储器地址。
- **PARAMETER(命令参数):** 该参数就是一个完整的 14443 命令或卡片定义的其他命令。模块在执行 SEND_RECEIVE 命令的时候,会自动计算该部分的 CRC,并连同 SOF、本数据、CRC、EOF(完整帧)发送给卡片然后等待卡片的

应答，等待的超时界限就是参数“等待延时”指定的。（请参考 ISO14443 标准相关文档中关于 TR0 的定义）。

- **CHECKSUM(校验和):** 命令校验和是从 **COMMAND**(命令代码)开始的所有字节的**带进位位**的累加和的**反码**。初始值为 **0x78**，进位位为 **0**。

2. 上位机接收命令执行结果

数据头 + 长度 + 长度校验 + 命令代码 + 执行状态 + [数据] + 校验和
HEAD+LENGTH+ LEN_CHK +COMMAND+STATUS+[DATA]+ CHECKSUM

- **HEAD(数据头):** 数据头固定为 0xA6。
- **LENGTH(长度):** 整个数据帧中除了数据头、长度、长度校验以外的所有数据的字节数（包括数据校验和）。该参数一个字节。
- **LEN_CHK (长度校验):** 数据长度字节取反。该参数一个字节。
- **COMMAND(命令代码):** 模块返回数据所对应的命令的代码。该值应该和最近一次发送给模块的命令帧中的“命令代码”参数一致。详细的命令说明在第五部份介绍。该参数一个字节。
- **STATUS(执行状态):** 该命令执行的结果。如果命令执行成功则状态为“0”。这个命令状态只标志模块的执行命令和通讯的状态，并不标志卡片的实际状态，卡片操作是否成功要在卡片的返回数据中表示。其他的错误代码在第五部分介绍。该参数一个字节。
- **DATA(数据):** 只有 SEND RECEIVE 以及 GET VERSION 命令返回数据。对于 SEND RECEIVE 命令该部分的数据就是卡片的应答数据，不包括 14443 协议中的 CRC。对于 GET VERSION 命令该部分的数据就是模块的版本信息。
- **CHECKSUM (校验和):** 返回数据校验和是从“命令代码”开始的所有字节的**带进位位的累积和**的**反码**。初始值为 **0x78**，进位位为 **0**。

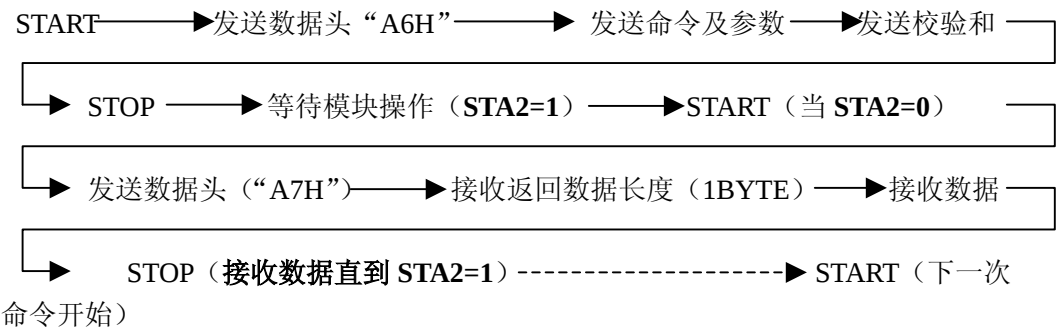
3. 数据校验方式

接收和发送两个过程中的数据校验方式都是采用带进位的字节累加和取反的方法计算。参加校验和运算的数据是数据帧中除了“HEAD”、“LENGTH”、“LEN_CRC”以外的所有数据。校验和的初始值为“0x78”。校验和的计算方法代码在附录中有说明。

4. TWI 接口的特殊说明

对于 TWI 接口上位机发送命令给模块的协议和 SPI 以及 UART 是一致的。但是由于模块被定义为 TWI 的从机，所以在命令执行结束不能主动发送数据给上位机。参考 I2C 协议的要求在上位机读取命令执行结果前需要先发送一个字节的命令头给

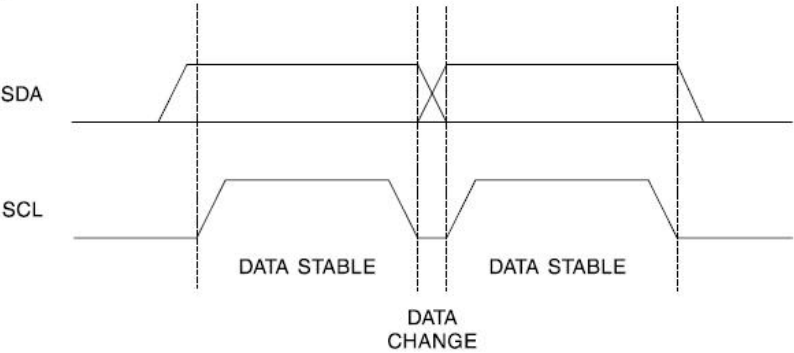
模块，在此命令头发送完毕（接收到来自模块的 ACK 信号），开始读取数据。完整的通讯过程可以描述为：



TWI 的数据物理层协议是参考 I2C 协议定义的。每个字节 9 个时钟。数据在 SCL 的上升沿有效。接收方每接收一个字节（8 BIT）的数据，需要在第九个时钟给发送方一个 ACK 应答。ACK=0 表示接收正确。上位机发送命令给模块，每个字节结束模块给上位机一个 ACK。上位机读取模块的执行结果时，每读取一个字节就给模块一个 ACK，读取最后一个字节时，上位机需要给模块一个反向的 ACK（NACK，逻辑“1”），然后以 STOP 结束通讯。**在用 TWI 接口时，模块返回的数据中没有数据头：0xA6（参考 figure19-5）**

TWI 接口的数据时序如下：

Data Validity



Start and Stop Definition

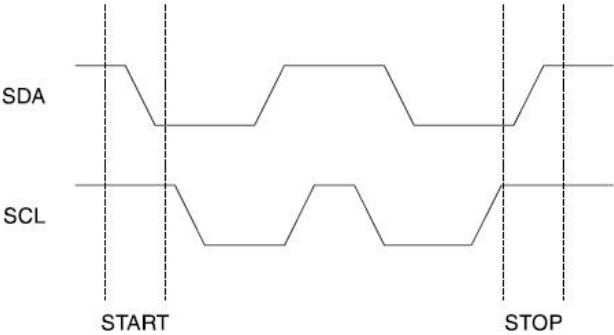
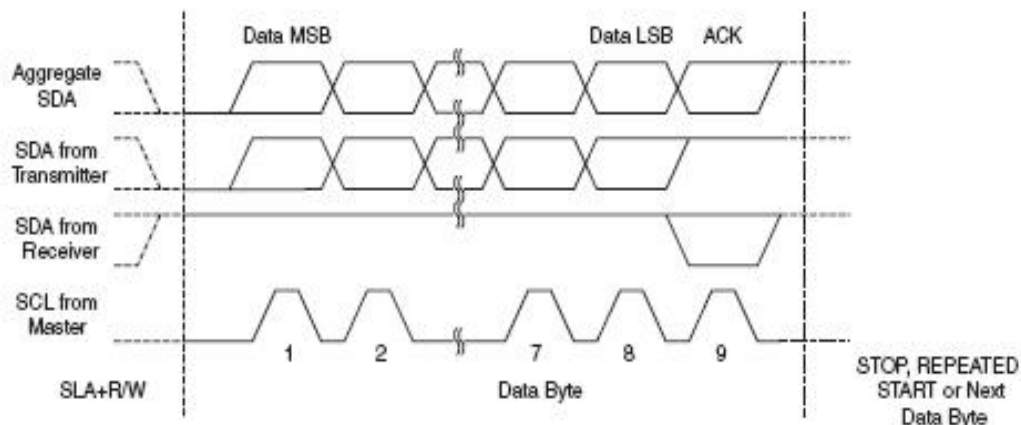


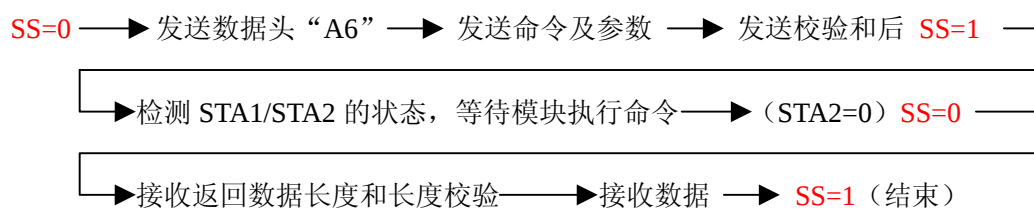
Figure 19-5. Data Packet Format



5. SPI 接口说明:

模块的 SPI 接口采用 MODE 0 的从机模式。SPI 的数据在时钟的上升沿有效。空闲状态时钟信号为低电平。在 SPI 模式从机模式下，模块也不能主动发送命令结果给上位机，只会在命令执行结束，将 STA2 置低，通知上位机发送时钟并取走数据。上位机通过 SS 管脚对模块进行片选。在 SPI 模式下，通讯不再像 TWI 那样需要 START/STOP 信号做开始和停止位，也不用在每个字节的末尾加 ACK 位。

通迅过程描述如下:



五. 命令与应答

下列的命令和返回值是以串行通讯协议为样板给的例子。在 TWI 模式下，模块返回值里面没有前面的 HEAD 部分。模块支持的命令列表如下表：

COMMAND	WAIT TIME	PARAMETER	RESPONSE	OTHER
0x10	5	NONE	STATUS	握手命令
0x11	卡片决定	卡片命令	卡片应答信息	卡片通讯
0x12	5	NONE	STATUS	关场
0x13	5	NONE	STATUS	开场
0x14	0x0A/0B	=WAIT TIME	STATUS	选择卡片协议
0x15	TIME	NONE	STATUS	蜂鸣器鸣叫
0x16	5	NONE	Version info	获取模块版本
0x17	5	NONE	Card info	自动检测卡
0x18	卡片决定	CPU 卡命令	卡片应答信息	CPU 卡通讯
0x19	ADDRESS	DATA	STATUS	写 EEPROM
0x1A	ADDRESS	LENGTH	DATA	读 EEPROM
0x1B	RATE	=WAIT TIME	STATUS	设置波特率
0x1C	5	NONE	DATA	获取协议信息
0x1D	5	MIFARE ONE COMMAND	MIFARE RESPONSE	Mifare one 卡读写
0x1E	5	读身份证 ID	DATA	读取身份证 ID 号
0x1F	5	PASM 卡	DATA	PSAM 卡通讯
0x20	5	执行 PASM 命令文件	PASM RESPONSE	批量执行 PSAM 交易命令
0x21	5	设置 7816 协议类型	RESPONSE	

以下为详细的命令说明

1. 命令说明

● 0x10: 握手命令 (HAND SHAKE)

该命令实现上位机和模块的握手操作。没有任何实际的功能，只是用来测试模块和上位机的通讯是否正常。命令没有参数，调用该命令在命令帧中的“等待延时”可以设置为“1-5”。下面的 16 进制代码为完整的关场命令的数据帧内容：

```
A6 03 FC 10 02 75
HEAD=A6
LENGTH=03
LEN_CHK =FC
COMMAND=10
```

WAITTIME=02

CHECKSUM =75

模块返回的数据为:

A6 03 FC 10 00 77

HEAD=A6 (如果是 UART 模式下, 模块返回 3 个字节的数据头)。

LENGTH=03

LEN_CHK =FC

COMMAND=10

STATUS=00

CHECKSUM =77

- **0x11: 发送接收 (SEND RECEIVE)**

该命令实现上位机和卡片的数据交换操作, 上位机发送给卡片的命令在参数 PARAMETER(命令参数)中, 其中不包括 ISO14443 规定的 CRC 校验字节。模块在发送这些信息前会自动计算 CRC。调用该命令的时候要特别注意 WAIT TIME(等待延时)的设置。该参数为一个字节。模块发送命令以后等待卡片的应答的超时界限就是这个参数乘以“19.3”毫秒。应当严格按照不同型号卡片的技术文档的要求, 设置这个参数。下面是 ISO14443 TYPE B -3 中的 REQB 命令的实际数据帧内容供参考:

A6 06 F9 11 05 05 00 00 F7

HEAD=A6

LENGTH=06

LEN_CHK =F9

COMMAND=11

WAIT TIME=05

PARAMETER="050000" (标准卡片指令, 参考卡片资料和 14443 标准)

CHECKSUM =F7

模块返回信息:

A6 0F F0 11 00 50 00 00 00 01 FF FF FF 64 00 30 51 3F

HEAD=A6

LENGTH=0F

LEN_CHK =F0

COMMAND=11

STATUS=00

DATA=" 50 00 00 00 01 FF FF FF 64 00 30 51" (具体的数据意义请参考卡片和 ISO14443 标准文档)。

CHECKSUM =3F

- **0x12: 关场命令 (CLOSE FIELD)**

该命令使模块关闭射频场。该命令和开场命令配合可以实现给卡片硬复位的功能, 另外关场以后模块的功耗会大大降低。命令没有参数, 调用该命令在命令帧中的“等待延时”可以设置为“1-5”。下面的 16 进制代码为完整的关场命令的数据帧内容:

A6 03 FC 12 05 70

HEAD=A6

LENGTH=03

LEN_CHK =FC

COMMAND=12

WAIT TIME=05

CHECKSUM =70

模块返回的数据为:

A6 03 FC 12 00 75

HEAD=A6

LENGTH=03

LEN_CHK =FC

COMMAND=12

STATUS=00

CHECKSUM =75

- **0x13: 开场命令 (OPEN FIELD)**

该命令使模块打开射频场, 准备读写卡片。命令没有参数, 调用该命令在命令帧中的“等待延时”可以设置为“1-5”。下面的 16 进制代码为完整的开场命令的数据帧内容:

A6 03 FC 13 05 6F

HEAD=A6

LENGTH=03

LEN_CHK =FC

COMMAND=13

WAIT TIME=05

CHECKSUM =6F

模块返回的数据为:

A6 03 FC 13 00 74

HEAD=A6

LENGTH=03

LEN_CHK =FC

COMMAND=13

STATUS=00

CHECKSUM =74

- **0x14: 选择 ISO14443 协议 (SELECT PROTOCL)**

该命令选择模块与卡片之间通讯需要符合的标准类型。ISO14443 目前有两个主要的分支, 一个是 TYPE A 标准 (MIFARE 1 以及兼容卡片符合的标准)。另一个是 TYPE B 标准 (ATMEL,ST 等公司的卡片遵循的标准, 主要的卡片有 AT88RF020,AT88SC0104CRF 系列, ST_SRI176、ST_SRIX4K 等)。使用该命令设置卡片的协议类型, 针对某一种类型的卡片的操作, 只需要在操作前设置一下该命令就可以了。如果要同时操作多种协议的卡片, 则在更换协议的时候, 需要调用这个命令。调用该命令的时候 WAIT TIME 参数作为一个命令参数使用。WAIT TIME=0A (进入 TYPE A 模式), WAIT TIME=0B (进入 TYPE B 模式)。下面的 16 进制代码为完整的选择 TYPE A 协议命令的数据帧内容:

A6 04 FB 14 0A 0A 5F

HEAD=A6

LENGTH=04

LEN_CHK =FB
COMMAND=14
WAIT TIME=0A
PARAMETER="0A"
CHECKSUM =5F

模块返回的数据为:

A6 03 FC 14 00 73
HEAD=A6
LENGTH=03
LEN_CHK =FC
COMMAND=14
STATUS=00
CHECKSUM =73

● **0x15: 蜂鸣器操作命令 (BEEP OPTION)**

该命令使模块的 BEEP 管脚 (PIN18) 发送一个制定宽度的脉冲, 可以驱动外部的蜂鸣器。调用该命令在命令帧中的 WAIT TIME(等待延时)参数表示 PIN13 置低的时间。可以设置为 “0-255”。下面的 16 进制代码为完整命令的数据帧内容:

A6 03 FC 15 FF 72
HEAD=A6
LENGTH=03
LEN_CHK =FC
COMMAND=15
WAIT TIME=FF
CHECKSUM =72

模块返回的数据为:

A6 03 FC 15 00 72
HEAD=A6
LENGTH=03
LEN_CHK =FC
COMMAND=15
STATUS=00
CHECKSUM =72

● **0x16: 获取版本信息 (GET VERSION)**

该命令返回当前使用模块的版本信息。命令没有参数, 调用该命令在命令帧中的 “等待延时” 可以设置为 “1-5”。下面的 16 进制代码为完整的命令帧内容:

A6 03 FC 16 05 6C
HEAD=A6
LENGTH=03
LEN_CHK =FC
COMMAND=16
WAIT TIME=05

CHECKSUM =6C

模块返回的数据为:

A6 2F D0 16 00 53 74 61 6E 74..... 00 66 A4

HEAD=A6

LENGTH=24

LEN_CHK =D0

COMMAND=16

STATUS=00

DATA=" 53 74 61 6E 74..... 31 36"

CHECKSUM =A4

完整的版本信息是一串 ASCII 码字符串

"Standard ISO14443 RF Reader,V1.02,2009,04,16"

● 0x17: 检测卡 (DETECT CARD)

该命令每执行一次,读卡器将自动开场,并发送监测卡片命令。若发现卡片就锁定卡片,并返回卡片的 ID 信息和协议字节。这个命令可以自动检测 TYPE A 和 TYPE B 的卡片。监测到卡片后模块将返回数据,并保持开场状态。若没有发现卡片,模块将自动关场,并返回错误信息代码。调用该命令的时候要特别注意 WAIT TIME(等待延时)的设置:该参数为从开场到发送 REQB 命令的时间,也就是卡片从上电复位到可以接受命令的时间,单位为毫秒。具体采用多长的延时时间,可以参考卡片的技术文档。一般设置为 05 对大多数卡片就足够了。

A6 03 FC 17 05 6B

HEAD=A6

LENGTH=03

LEN_CHK =FC

COMMAND=17

WAIT TIME=05

CHECKSUM =6B

返回信息为:

● AT88SC6416CRF 卡

A6 10 EF 17 00 0B 50 00 00 00 01 FF FF FF 60 00 30 51 2E

HEAD=A6

LENGTH=10

LEN_CHK =EF

COMMAND=17

STATUS=00

DATA=" 0B 50 00 00 00 01 FF FF FF 60 00 30 51"

TYPE B

PUPI

APP

BYTE S

CHECKSUM =2E

● MIFARE ONE S50

A6 0C F3 17 00 0A 04 00 A6 A2 FA 69 97 08 15

HEAD=A6

LENGTH=0C

```

LEN_CHK =F3
COMMAND=17
STATUS=00
DATA=" 0A 04 00 A6 A2 FA 69 97 08 "
          TYPE A          ID          BCC
CHECKSUM =15

```

● **0x18: 标准 CPU 卡操作命令 (CPU CARD OPTION)**

该命令只对于操作符合 ISO14443 TYPE B/A 标准的 CPU 卡，双界面卡操作有效。在锁定卡片以后，调用该命令实现和 CPU 卡的数据交互。该函数自动在每个命令的前面加上 0A01 两个字节的头，对去一般的逻辑加密卡 AT88SC6416CRF, AT88RF020, MIFARE I 等卡片该命令无效。下面以取随机数命令为例描述关于 CPU 卡通讯的命令的执行过程

```

A6 08 F7 18 05 00 84 00 00 08 DD
HEAD=A6
LENGTH=08
LEN_CHK =F7
COMMAND=18
WAIT TIME=05 (延时时间为 FWT×5)
PARAMETER=" 00 84 00 00 08"
CHECKSUM =DD
模块返回的数据为:
A6 0D F2 18 00 ** ** ** ** 90 00 **
HEAD=A6
LENGTH=0D
LEN_CHK =F2
COMMAND=18
STATUS=00
DATA="** ** ** ** 90 00"(8 字节随机数+CPU 卡状态 "9000")
CHECKSUM =6E

```

● **0x19: 写 EEPROM (WRITE EEPROM)**

该命令向模块中的 EEPROM 写入数据。注意：模块中的 EEPROM 的容量为：256 字节。在实际使用的时候，最前面的一个字节不要使用。另外该命令的延时时间按照下面的单位进行计算。每个字节 4mS。命令中的 WAIT TIME 参数在这里作为写入字节的地址使用。

```

A6 0A F5 19 00 12 34 56 78 00 00 00 59
HEAD=A6
LENGTH=0A
LEN_CHK =F5
COMMAND=19
WAIT TIME=00 (从 00H 开始写入数据)
PARAMETER=" 12 34 56 78 00 00 00"
CHECKSUM =59
模块返回的数据为:
A6 03 FC 19 00 6E

```

HEAD=A6
LENGTH=03
LEN_CHK =FC
COMMAND=19
STATUS=00
CHECKSUM =6E

● **0x1A: 读 EEPROM (READ EEPROM)**

该命令从模块的 EEPROM 中读取数据。注意：模块中的 EEPROM 的容量为：256 字节。在实际使用的时候，最前面的一个字节不要使用。另外该命令的延时时间按照下面的单位进行计算。每个字节 1 mS。命令中的 WAIT TIME 参数在这里作为字节的地址使用。下面的 16 进制代码为完整的读 EEPROM 命令的数据帧内容：

A6 04 FD 1A 00 04 69
HEAD=A6
LENGTH=04
LEN_CHK =FD
COMMAND=1A
WAIT TIME=00（从 00H 开始写入数据）
PARAMETER=" 04"（读取 4 个字节的数据）
CHECKSUM =69
模块返回的数据为：
A6 07 F8 1A 00 12 34 56 78 58
HEAD=A6
LENGTH=07
LEN_CHK =F8
COMMAND=1A
STATUS=00
DATA=" 12 34 56 78"
CHECKSUM =6E

● **0x1B: 设置卡片通讯波特率 (LIMIT RATE)**

该命令限制卡片与基站之间的通讯波特率。在相同的条件下建议使用 106K 的波特率，可以获得更高的系统稳定性。

A6 04 FB 1b 0F 0F 69
HEAD=A6
LENGTH=04
LEN_CHK =FD
COMMAND=1B
WAIT TIME=0F（波特率设置值 0x0F 表示使用 106K 进行收发数据）。
PARAMETER=0F（波特率设置值 0x0F 表示使用 106K 进行收发数据）。
CHECKSUM =XX
模块返回的数据为：
A6 03 FC 1B 00 XX
HEAD=A6
LENGTH=03

LEN_CHK =FC
 COMMAND=1B
 STATUS=00
 CHECKSUM =XX

● **0x1C: 获取卡片通讯设置 (GET RATE)**

该命令读取卡片与基站之间相互通讯的参数设置。该命令返回 5 个字节的
 状态信息。表明卡片与基站之间通讯所需要的协议以及相关的设置信息

DATA0: 通讯协议类型, 通过该字节可以判断卡片通讯协议的类型。

0x0A----- TYPE A 协议

0x0B----- TYPE B 协议

DATA1: 通讯速率, 一个字节。高位四个 BIT 保留没有使用,BIT2,3 表示
 卡片的接收波特率, BIT1,0 表示卡片的发送波特率。

BIT7,BIT6	BIT5,BIT4	BIT3,BIT2	BIT1,BIT0	(二进制)
00	00	接收速率	发送速率	
		00	00	106K
		01	01	212K
		10	10	424K
		11	11	848K

DATA2: 协议类型。

01: CPU 卡

00: 非 CPU 卡

DATA3: 卡片的数据缓冲区大小。一个字节的十六进制数据。

DATA4: FWT。卡片需要的最小超时限制, 一个字节的十六进制数据。

卡片处理命令的时间, 实际时间为 FWT×19.3Ms.

A6 03 FC 1C 05 66

HEAD=A6

LENGTH=03

LEN_CHK =FC

COMMAND=1C

WAIT TIME=05

PARAMETER 无

CHECKSUM =XX

模块返回的数据为:

A6 05 FA 1C 00 11 22 33 44 55 XX

HEAD=A6

LENGTH=05

LEN_CHK =FA

COMMAND=1C

STATUS=00

DATA= 11 22 33 44 55

CHECKSUM =XX

● **0x1D: MIFARE ONE 系列卡片读写命令 (SEND/RECEIVE MIFARE ONE)**

该命令用于完成 MIFARE ONE 系列卡片的读写操作。下面以 MIFARE
 ONE 的 HALT 命令为例说明该命令的使用方法:

A6 04 FB 1D 05 50 15
 HEAD=A6
 LENGTH=04
LEN_CHK =FB
 COMMAND=1D
 WAIT TIME=05（等待延时）。
 PARAMETER=50（mifare one 卡的 HALT 命令）。
CHECKSUM =15
 模块返回的数据为：
 A6 03 FC 1D 00 6A
 HEAD=A6
 LENGTH=03
LEN_CHK =FC
 COMMAND=1D
 STATUS=00
CHECKSUM =6A

MIFARE ONE 卡片专用命令列表：

命令	代码	参数	返回	备注
AUTH	60/61	BLOCK ADDRESS+KEY(6 BYTES)	STATUS	认证密钥
READ	30	BLOCK NUM	STATUS	读取数据块内容
WRITE	A0	BLOCK ADDRESS+DATA(16 bytes)	STATUS	更新数据块数据
DECREMENT	C0	BLOCK ADDRESS+DATA(4 bytes)	STATUS	电子钱包扣值
INCREMENT	C1	BLOCK ADDRESS+DATA(4 bytes)	STATUS	电子钱包充值
RESTORE	C2	BLOCK ADDRESS+DATA(4 bytes)	STATUS	Block-----buffer
TRANSFER	B0	BLOCK ADDRESS	STATUS	Buffer-----block
HALT	50	NC	STATUS	卡片挂起

详细的 MIFARE ONE 卡片的读写操作请参考 MIFARE ONE 卡片读写操作专题和例程。对 MIFARE ONE 卡片操作前可以调用 DETECT CARD(0x17)命令，检测并锁定卡片。

● **0x1E：读取二代身份证芯片 ID 号命令**

该命令用于读取二代身份证卡的 ID 号。

A6 06 F9 1E 05 05 00 00 5F
 HEAD=A6
 LENGTH=06
LEN_CHK =F9
 COMMAND=1E
 WAIT TIME=05（波特率设置值 0x0F 表示使用 106K 进行收发数据）。
 PARAMETER=05 00 00（TYPE B 卡 ATQB 命令）。
CHECKSUM =5F
 模块返回的数据为：
 A6 37 C8 1E 00 50 00 00 00 00 ** 03 ** 05 00 ** ** 20 ** 20 ** ** 24 ** 78

00 01 ** ** 01 06 03 14 00 00 00 00 06 ** 10 70 ** 4B ** 8B ** ** AA ** 86
18 ** 93 ** ** **

HEAD=A6

LENGTH=37

LEN_CHK =C8

COMMAND=1E

STATUS=00

DATA=" 50 00 00 00 00 ** 03 ** 05 00 ** ** 20 ** 20 ** ** 24 ** 78 00

身份证卡 ID 号

01 ** ** 01 06 03 14 00 00 00 00 06 ** 10 70 ** 4B ** 8B ** ** AA **

86 18 ** 93 ** ** ** "

(* 号部分为隐去的身份信息。返回的数据信息里面从第 13 个字节开始的 8 个字节的数据是身份证卡片的 ID 号)。

CHECKSUM =74

2. 返回状态码

模块返回的数据帧中参数 STATUS(执行状态)表示模块执行命令的状态。所有的命令代码如下：

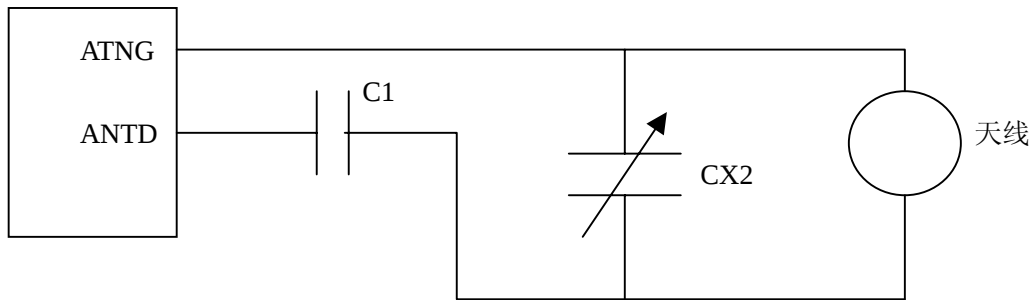
- 00H 操作成功
- 01H-77H 监测 TYPE A 卡片的时候出现冲突。
- 78H 接收卡片数据，CRC 校验错。
- 80 H 读写卡片时，没有开场
- 81H 卡片无应答
- 82H 卡片返回数据错误
- 83H 调制常开
- 84H 通讯命令校验和错误
- 85H 未知错误
- 86H 命令长度错误
- 87H EEPROM 操作错误
- 88H ISO14443 协议 CPU 卡通讯错误（针对 0x18 命令）
- 89H MIFARE ONE 卡片操作错误（针对 0x1D 命令）

附录

1. 参考天线设计

- 天线驱动原理

模块天线采用并联谐振的驱动方式，电线的阻抗为标准的 50 欧姆。天线驱动电路的原理如下图：



其中：

C1=47pF

CX2 需要根据不同的天线选取不同的参数，使 CX2 与天线组成的谐振电路的固有频率为 13.56MHz 即可。

计算公式：

$$f_r = \frac{1}{2\pi\sqrt{LC}}$$

- 天线制图笔记

1. 天线原理图设计

在实际应用过程中天线的电感是很难准确计算的，所以会多并联几个电容或设计一个可调的电容进行微调。或采用固定电容和可调电容并联搭配的设计方法，提高天线的设计成功率。

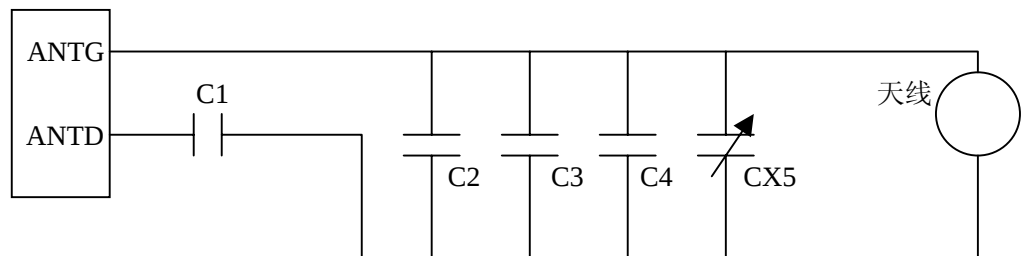


图 (F1-1)

其中 C2,C3,C4,CX5 并联在一起，可以等效为一个大电容

$$C=C2+C3+C4+CX5$$

在实际使用中，这几个电容不是必须都焊接的。可以采用选焊的方法组合出需要的谐振电容 C。

2. 天线 PCB 制图

设计读卡器天线主要是天线线圈的绕制。可以采用双绞线的绕制方法，也可以采用平行线的绕制方法。总体上讲没有太多的限制。一个重要的指标就是天线的电感值不要超过 **2uH**。天线的周边如果有金属物或靠近铁板，覆铜的 PCB 的情况下，会影响天线的电感值，和射频场的频率。一般当天线（电感线圈）靠近金属物时电感会变大。所以在设计天线的时候，如果已知天线装配的位置存在金属物或接近金属覆铜板等导电材料，应适当的减少线圈的匝数。我们尝试过将天线绕制在有覆铜的应用电路的外周。只要天线的匝数选择合适，最里层线圈和覆铜外沿有一定的距离（**>5mm**）还是可以得到满意的读卡效果的。

在实际应用产品的设计中，很多时候天线是作为外接的一个部件设计在一个单独的天线板上的。对于这种应用我们的建议是：模块支持 50 欧姆匹配的天线，如果天线需要很长的引出线，建议使用 50 的同轴电缆。另外尽量将与天线相关的匹配电容 C1,C2,C3,C4,CX5 设置在天线端。这样可以有效的提高系统的工作稳定性。

关于天线的 Q 值，这里不过多的介绍有关 Q 值的问题。需要明白的一个规律就是 Q 值越高，天线的射频场强越高，理论上读卡的距离也越远（提供的能量比较大）。但是 Q 值另一方面的影响是随着 Q 值的升高，天线允许的带宽在变窄。也就是说一个 Q 值很高的天线对频率的选择性越强。而我们需要读取的卡片往往会因为半导体和制卡等环节的工艺问题而出现频率的偏差。Q 值过高的天线会出现挑卡或断层（卡片只在某个距离上可以读取到）等现象发生。只有使产品在读卡距离和卡片的兼容性方面达到一个合理的平衡点才能设计出性能优越的读写卡设备。

空心线圈的电感量计算公式如下：

$$L=(0.01*D*N*N)/(L/D+0.44)$$

其中

线圈电感量 L 单位：微亨

线圈直径 D 单位：cm

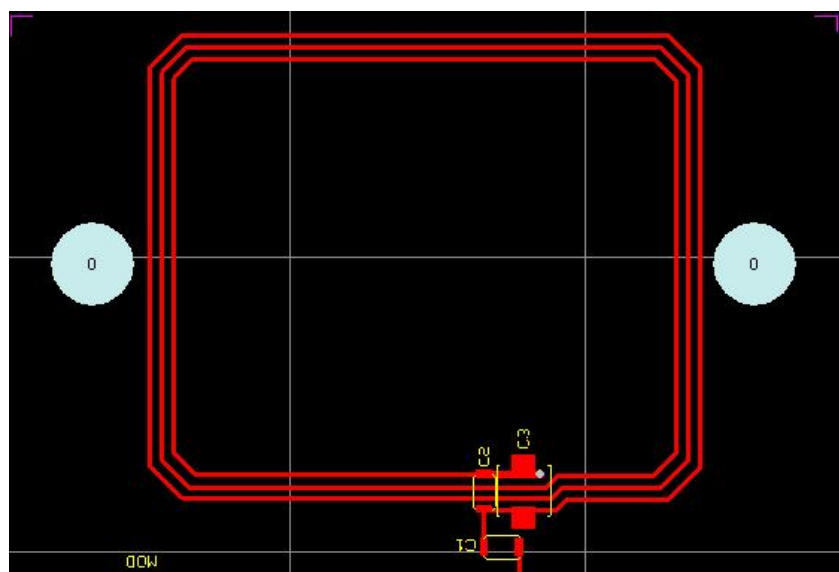
线圈匝数 N 单位：匝

线圈长度 L 单位：cm

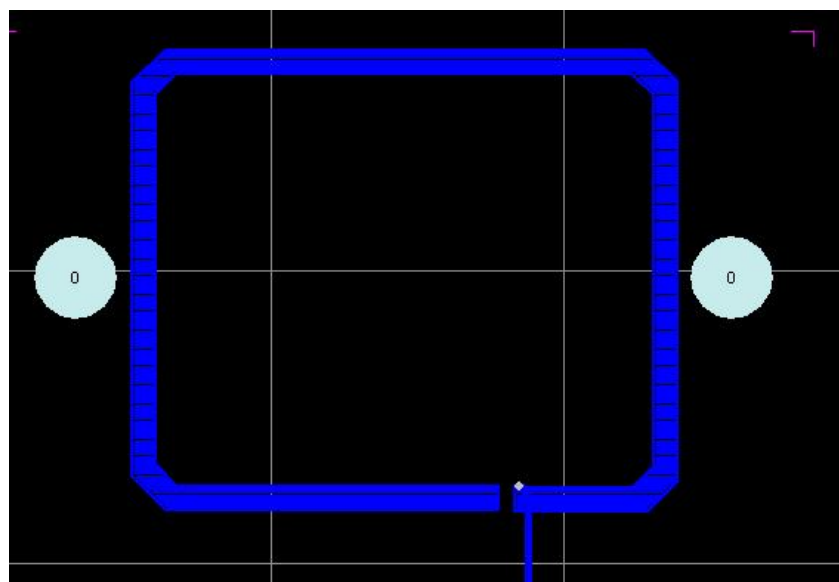
我们在实际设计线圈的时候不可能按照公式精确的计算天线的电感值，只能从公式中了解到天线的电感和天线的截面，线圈的匝数，所用导线的线径成正比例的关系的。要得到同样的电感线圈，在使用导线相同的情况下，大截面的线圈的绕制圈数要小于小截面的天线。

下面给出几个我们设计的天线样板的实例希望对用户设计天线有所帮助。

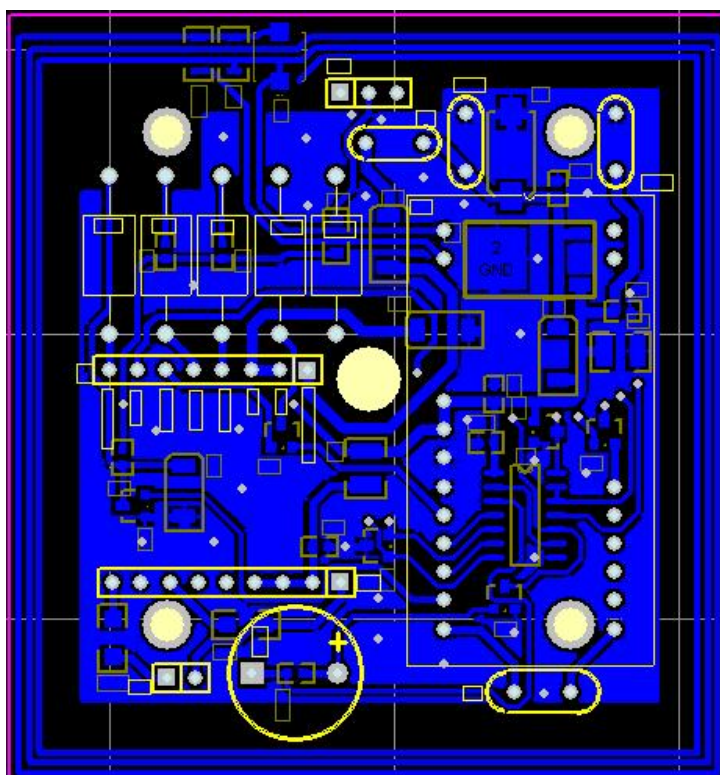
天线 1 正面 (48mm×40mm) 标准的天线绕制方式



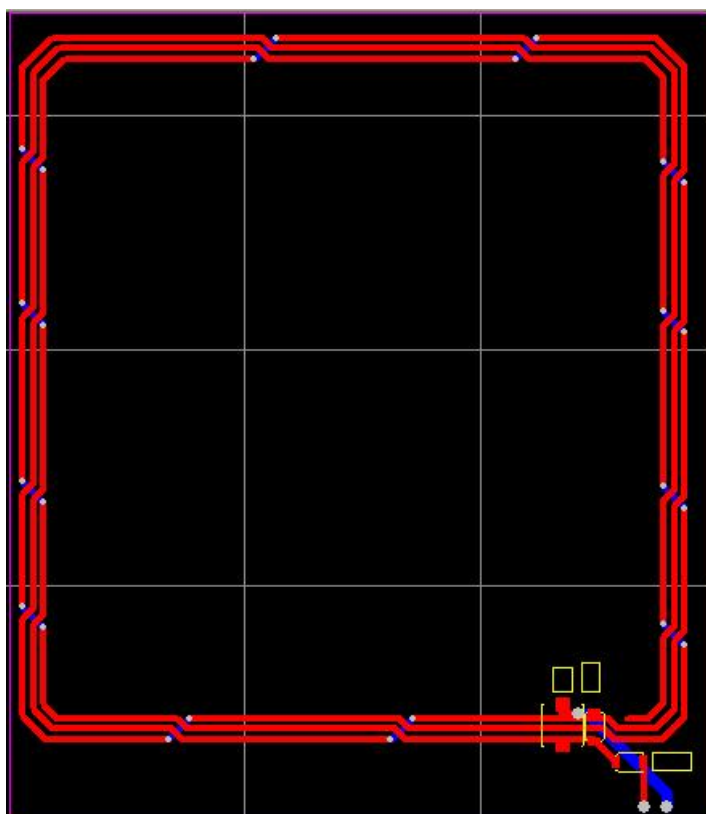
天线 1 反面



天线 2 正面（63mm*66mm）平行绕制带覆铜模式



天线 3（72mm*76mm）双绞线模式



以上 3 个型号的天线都是在实际应用中成功的天线实例。在实际设

计时应该留意，我们给每个线圈都添加了一圈辅助绕组（最里层的一圈线圈是不封口的）。这样可以有效的提高读写卡的稳定性。建议用户也要在设计自己的天线的时候也要加上最里层的一圈辅助绕组。

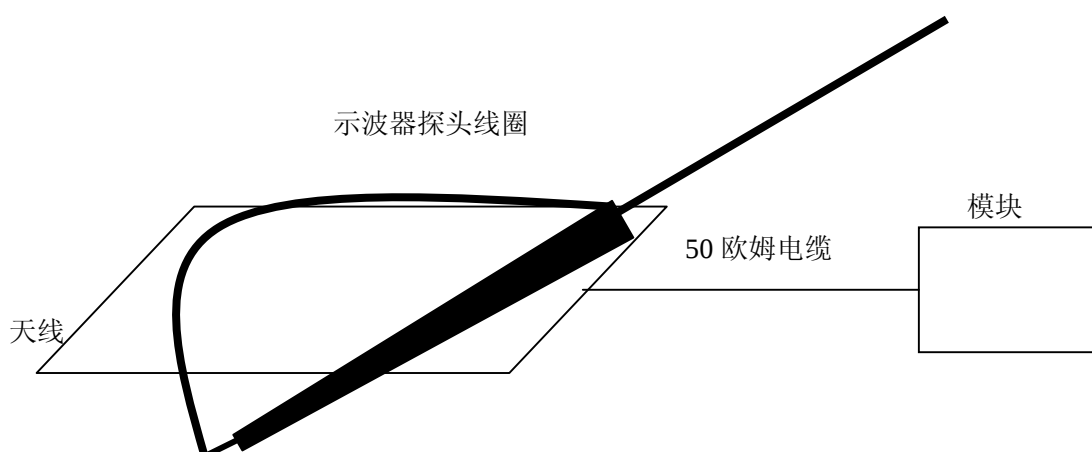
● 天线调试笔记

由于我们绘制天线的时候无法准确的得出所绕制的线圈的电感值，所以与线圈匹配的谐振电容的选取就需要在拿到天线 PCB 板后进行细致的调试以获取准确的电路参数。

利用并联谐振电路的鉴频的原理可以很容易的对线圈的谐振电容进行调整。在同等条件下，当谐振电路的固有频率和信号频率相同的时候，天线向外辐射的射频场的场强最大。

1. 调试工具

- a) 示波器：使用示波器将示波器的纵轴（电压）和横轴（时间）参数设置在合理的范围内，以能清楚的看到，射频信号为准。取一个表笔，将表笔的地线和探头合成一个回路（地线上的夹子夹住自己的探头，也可以找一个单芯的导线，围成和天线大小接近的圆圈，表笔的探头和地线分别和线圈的两端连接）。将探头环或线圈和要测试的天线放在一个同轴的平面上。如图：



- b) 无感改锥：使用无感改锥的目的是减少，金属部件对可调电容的参数影响。
- ### 2. 调试步骤：假设天线的原理图如前面所给的图（F1-1）所示。
- a) C1=30pF C2, C3, C4 不焊 CX5 焊接一个 0-30pF 的可调电容。
 - b) 给电路通电，给模块发送开场命令，让模块开始发送 13.56MHz 的天线驱动信号。这时调整示波器的 X/Y 轴的设置，可以在示波器上看到一个 13.56MHz 的正玄波。
 - c) 用无感改锥调整 CX5 的值，观察示波器上正玄波的变化。当正玄波的峰峰值达到相对的最大值时，停止调整 CX5。记录此时的峰峰值。
 - d) 在 C2 上焊接一个 30pF 的电容，然后重复 b—c 的操作。比较

两次测量的峰峰值，看加上 C2 这个电容以后，峰峰值是变大了，还是变小了。如果峰峰值变大了则在 C3 上再并联一个 30pF 的电容以后重复 b---d 的步骤。直到获得相对最高的峰峰值的那组参数为止。

- e) 如果再给天线电路并联电容不能使正弦波的峰峰值变高时。表明匹配的电容参数已经确定。记录能够产生最大峰峰值的那组电容的容值。测量 CX5 的实际电容量，和固定电容的容值相加就可以获得实际需要的谐振电容的大小了。
- f) 实际由于很难购买到与测量获得的电容量相同的固定电容，可以采用多个电容并联的方法用多个不同容值的电容搭配出所需要的谐振电容来。这里需要注意的就是天线电路中的 C1,C2,C3,C4,CX5 的耐压值建议使用>100V 的高耐压电容。
- g) 注意：理论上在峰峰值达到最大的时候天线的谐振频率与模块的驱动信号频率相同。此时射频的场强最大。但是为了达到理想的卡片兼容性，我们往往会使天线的谐振频率略微的偏移 13.56MHz。具体的偏移量需要根据实际读写卡的距离和稳定性确定。

2. MIFARE ONE 卡片操作说明

- 卡片识别

调用 0x17 命令，自动检测卡片。

1)。发送命令给模块：A6 03 FC 17 05 6B

命令码：0x17

WAITTIME: 05

2)。模块返回：A6 0C F3 17 00 0A 04 00 F2 B5 0D 6B 21 08 18

命令码：0x17

状态码：0x00

协议类型：0x0A

卡片类型：04 00 S50 卡

02 00 S70 卡

卡片 ID 信息： F2 B5 0D 6B 21 08

- 密钥认证

MIFARE 卡（S50/S70）对卡片内的用户数据区按照扇区进行划分，S50 卡分为 16 个扇区，S70 卡为 128 个扇区。每个扇区又分为 4 个 BLOCK。其中 BLOCK0-BLOCK2 为用户数据区，最后一个 BLOCK3 保存该扇区的密钥和访问权限限制。每个扇区拥有 A/B 两组密钥。

1)。发送命令给卡片：A6 0B F4 1D 05 60 00 FF FF FF FF FF FF 05

命令码：0x1D

WAITTIME: 0x05

卡片命令：0x60 00 FF FF FF FF FF FF

0x60: 认证 A 密钥。（0x61 认证 B 密钥）

0x00: 数据块（BLOCK）编号（认证数据块时该编号是数据块在整个卡片中的顺序号，例如：第五扇区的第 3 块 则数据块号为：0x13；第二扇区的第一块的数

据块号为：0x06)。

0Xff—ff：密钥内容（每组密钥由 6 个字节组成）。。

2)。返回数据：A6 07 F8 1D 00 D3 87 16 C7 31

命令码：0x1D

状态码：0x00（如果认证不成功，则会返回错误编码）。

卡片返回数据：D3 87 16 C7（卡片返回的 4 个自己的随机数，可以忽略）。

● 卡片数据读写

■ 读卡

1)。发送命令给卡片：A6 05 FA 1D 05 30 00 35

命令码：0x1D

WAITTIME：0x05

卡片命令：0x30 0x00

0x30：读卡命令

0x00：数据块的编号（MIFARE 卡的数据块编号是所有扇区数据块加在一起进行统一编号的。例如：第五扇区的第 3 数据块的数据块的编号为：0x13；第二扇区的第一数据块的数据块号为：0x06）。

2)。返回数据：A6 13 E7 1D 00 82 D6 49 69*****68 69 AE

命令码：0x1D

状态码：0x00（如果没有读到数据则返回其他错误代码）。

卡片数据：82 D6 49 69*****68 69（每个数据块为 16 个字节）。

写卡

1)。发送命令给卡片：A6 15 EA 1D 05 A0 01 22 22 22 ***** 22 22 A1

命令码：0x1D

WAITTIME：0x05

卡片数据：A0 01 22 22 22 ***** 22 22

0xA0：MIFARE 写卡命令

0x01：数据块的编号（如上面命令的说明）。

0x22 0x22 *****0x22 0x22：要写入数据块的数据（16 字节）。

2)。返回数据：A6 04 FB 1D 00 0A 60

命令码：0x1D

状态码：0x00（如果返回其他错误编码，便是写卡失败）。

卡片返回状态：0x0A（可以忽略）。

● 电子钱包初始化

MIFARE 卡可以用作电子钱包使用。卡片支持电子钱包交易用的充值和扣值命令。要使某一个数据块用作电子钱包就要按照电子钱包对数据格式的要求首先对该数据块进行初始化。电子钱包数据块的格式如下：

00000000 FFFFFFFF 00000000 01 FE 01 FE
余额 余额取反 余额 数据块号 块号取反 块号 块号取反
余额采用低字节在前的方式保存在 4 个字节中。

直接使用写卡命令将符合格式限制的数据写入指定的电子钱包数据块就完成该数据块的初始化工作。

● 电子钱包充值

完成电子钱包的冲值需要执行 MIFARE ONE 卡的两个命令

(INCREMENT, TRANSFER) 来共同完成。

1)。充值 (INCREMENT)

发送命令: A6 09 F6 1D 05 C1 01 0C 00 00 00 96

命令码: 0x1D

WAITTIME: 0x05

卡片命令: C1 01 0C 00 00 00

0xC1: 充值命令

0x01: 电子钱包编号

0x0C 0x00 0x00 0x00: 充值金额 (十进制 12, 低字节在前)。

返回: A6 03 FC 1D 00 6A

命令码: 0x1D

状态码: 0x00

2)。确认 (TRANSFE)

发送命令: A6 05 FA 1D 05 B0 01 B3

命令码: 0x1D

WAITTIME: 0x05

卡片命令: B0 01

0xB0: TRANSFER 命令编码

0x01: 电子钱包的数据块编号, 要和前面执行的 INCREMENT 命令中的数据块编号一致。

返回数据: A6 04 FB 1D 00 0A 60

命令码: 0x1D

状态码: 0x00

卡片状态码: 0x0A (可以忽略不做处理)。

● 电子钱包扣值

和完成电子钱包充值类似, 从电子钱包中扣值也需要两个 MIFARE 命令 (DECREMENT, TRANSFER) 来实现。

1)。扣值 (DECREMENT)

发送命令: A6 09 F6 1D 05 C0 01 0A 00 00 00 99

命令码: 0x1D

WAITTIME: 0x05

卡片命令: C0 01 0A 00 00 00

0xC0: 扣值命令 (DECREMENT)

0x01: 电子钱包所在数据块编号

0x0A 0x00 0x00 0x00: 扣除的金额 (十进制金额 10, 低位字节在前)。

返回数据: A6 03 FC 1D 00 6A

命令码: 0x1D

状态码: 0x00

2)。确认 (TRANSFE)

发送命令: A6 05 FA 1D 05 B0 01 B3

命令码: 0x1D

WAITTIME: 0x05

卡片命令: B0 01

0xB0: TRANSFER 命令编码

0x01: 电子钱包的数据块编号，要和前面执行的 INCREMENT 命令中的数据块编号一致。

返回数据: A6 04 FB 1D 00 0A 60

命令码: 0x1D

状态码: 0x00

卡片状态码: 0x0A (可以忽略不做处理)。

- 电子钱包余额查询

查询电子钱包的余额只要通过认证获取权限后用读卡命令把电子钱包的数据块数据读出，然后按照电子钱包的数据格式进行校验以后就可以得到实际的余额了。

MIFARE 卡的电子钱包中的余额使用 4 个字节的二进制数表示余额。低位字节在前。

3. CPU 卡操作说明

对符合 ISO14443 -1-2-3 标准的 CPU 卡，可以直接调用 0x18 命令。对于符合 PBOC2.0 标准的射频 CPU 卡/双界面卡，直接把 CPU 卡操作的命令发送给卡片就可以。用户不用考虑 ISO14443 对卡片通讯所规定的其他细节。下面举几个简单的命令实例说明对 CPU 卡的操作步骤。

- 卡片识别

调用 0x17 命令，自动检测卡片。这个命令可以自动识别符合 ISO14443TYPE A 或 TYPE B 标准的 CPU 卡/双界面卡。

1)。发送命令给模块: A6 03 FC 17 05 6B

命令码: 0x17

WAITTIME: 05

2)。模块返回: A6 11 EE 17 00 0B 50 75 62 53 A2 01 *****

命令码: 0x17

状态码: 0x00

协议类型: 0x0B

卡片 ATR 信息: 50 75 62 53 A2 01 ***** (具体的含义参考卡片资料)

- 获取波特率

用户检测到卡片 (DETECT CARD) 以后调用 0x1C 命令获取卡片通讯设置信息。该命令返回信息表明卡片的协议类型，发送波特率，接收波特率，是否是 CPU 卡，数据缓冲区 FIFO 的大小，等待超时限制等信息。用户可以根据这些信息对后面的读写操作做相应的配置。这里面特别重要的就是 FWT 的信息，这个数值将表明卡片可能的最大延时长度。当卡片执行某些特殊指令的时候可能会需要等待较长的时间 (这个和使用的卡片以及卡片里面运行的 COS 程序有关，个别卡片可能长达 2 S 以上)。

1)。发送命令给模块: A6 03 FC 1C 05 66

命令码: 0x1C

WAITTIME: 05

2)。接收数据: A6 07 F8 1C 00 0B 00 01 40 10 xx

命令码: 0x1C

状态码: 00

卡片类型: 0B

波特率: 00 (发送接收均采用 106K bps)

协议类型: 01 (是 CPU 卡)

FIFO 大小: 40 (64 BYTES)

FWT: 10 (16×19.3mS, 通常会使用这个时间参数作为以后跟卡片通讯的超时限制)。

- **设置通讯波特率**

大多数 CPU 卡的通讯波特率是可以设置的, 在相同条件下使用较低的波特率会取得更好的读写稳定性。模块可以支持 106K/212K/424K 波特率的卡片读写。当发现在某个较高的波特率下不能稳定的操作卡片时, 可以通过设置一个较低的波特率来获得稳定性的提高。大多数 CPU 都会使用 106K 作为默认的波特率。所以这个命令只对特殊的情况下才可使用。

- **取随机数**

取随机数是所有 CPU 卡都支持的命令, 这是启动一个密钥认证过程所必须的步骤。

1)。发送命令给模块: A6 08 F7 18 05 00 84 00 00 08 XX

命令码: 0x18

WAITTIME: 10 (fwt)

CPU 卡命令: 00 84 00 00 08 (取 8 个字节的随机数)

2)。接收数据: A6 0D F2 18 00 ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** 90 00 **

命令码: 0x18

状态码: 00

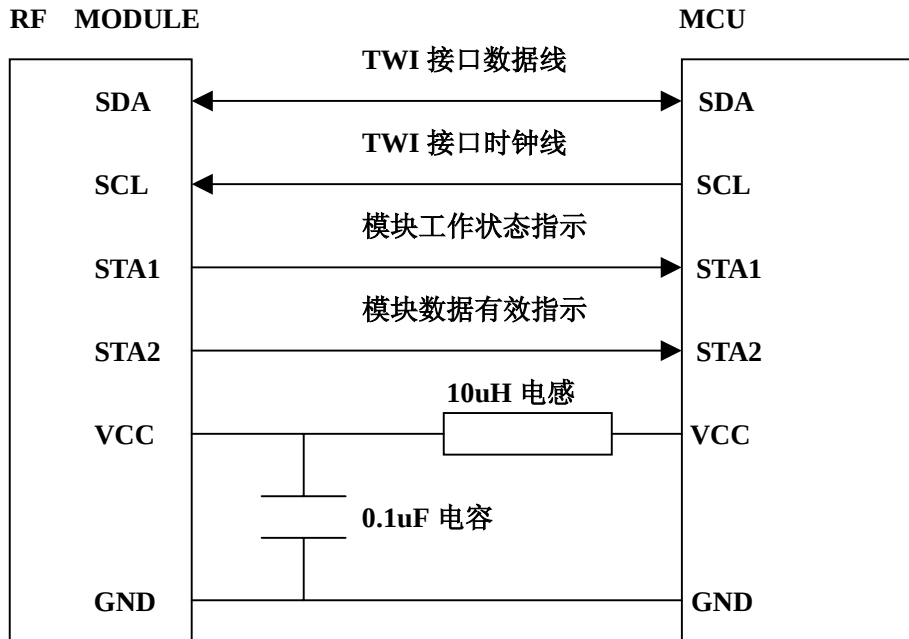
CPU 卡返回数据: ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** 90 00 (9000 表示命令执行成功)。

- **其他命令**

取完随机数以后, 就可以按照卡片的 COS 手册。进行其他的卡片操作。比如密钥认证, 打开文件, 读写文件等。

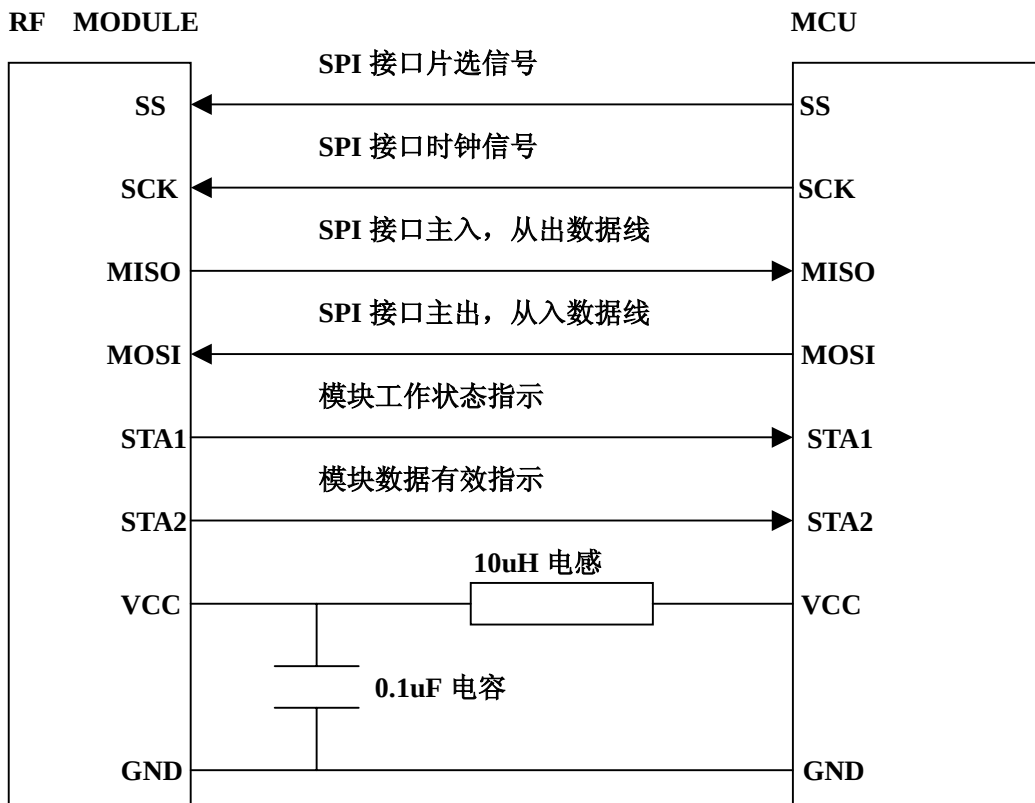
4. 标准应用电路设计

● TWI 接口电路

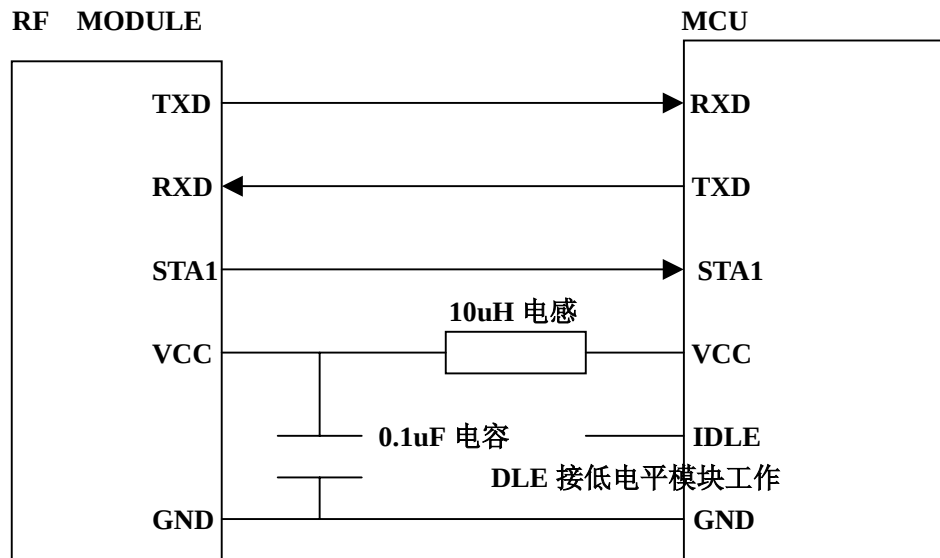


模块与 MCU 的电源采用一个 LC 滤波电路隔离一下。可以有效的避免相互的干扰。具体使用的隔离电感的数值要根据实际系统中干扰源来测定。该电感最好采用线绕的大功率电感。另外根据需要，模块的 IDLE 管脚可以直接接地（一直保持工作状态，功耗比较大）。也可以和上位机的 I/O 口连接，需要模块工作的时候输出“0”，如果不需要模块工作可以输出“1”使模块进入休眠模式。这样可以有效的降低系统的功耗。

● SPI 接口电路



- UART 接口电路



由于多数 MCU 串行通讯收发都可以设计为中断模式。标准的 UART 通讯为全双工模式。所以不用再连接 STA2 到上位单片机。只要检测 STA1 的电平变化和 UART 通讯是否超时即可完成 CPU 长超时操作的判断工作。

5. 通讯校验和计算方法

```
void Check_Sum(unsigned char *Data,short int Length,unsigned char *Checksum)
```

```
//DATA:原始数据
```

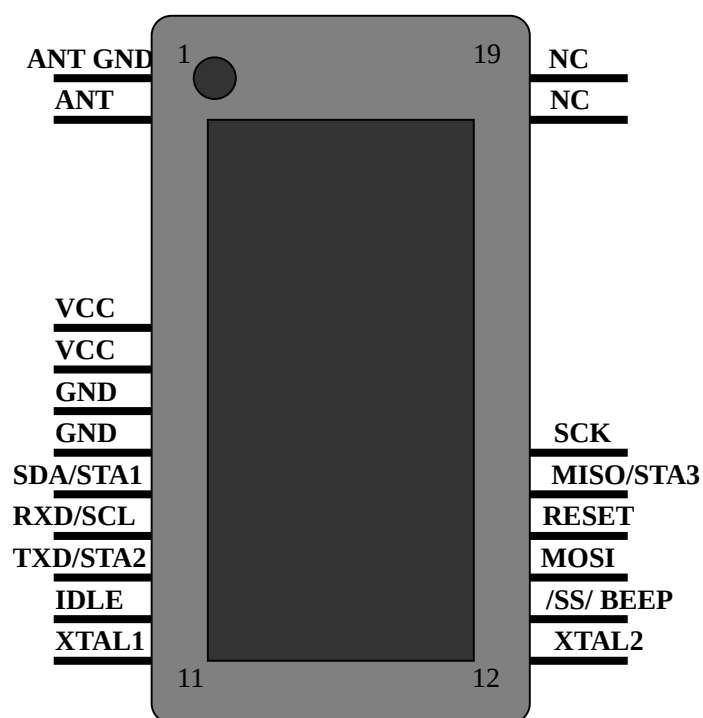
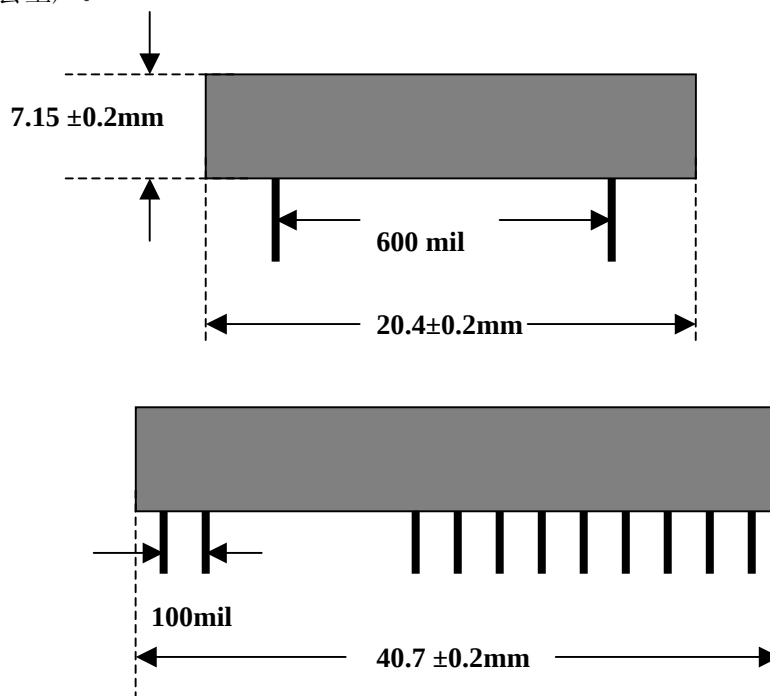
```
//LENGTH:计算校验和的数据长度
```

```
//CHECKSUM:计算结果
```

```
{
    int temp=120;
    char *Pb;
    for(int i=0;i<Length;i++)
    {
        temp+=*(Data+i);
        if((temp==255)||((temp==510)))
            temp=255;
        else
            temp%=255;
    }
    Pb=(char*)&temp;
    *Checksum=(*Pb)^0xff;
}
```

附早期 SL-13.56MOD 大模块外观示意图与管脚定义（机械图以下方给出的数据为准）：

说明：13.56M 射频模块我们有一大一小两种，小模块是在大模块的基础上改进生产工艺而来，两者只是体积和管脚定义有所区别，具体功能和命令集一致。今后我们将主推小模块，即 SL-1356MOD-SU。但考虑到以前使用大模块的客户，两种模块我们都会生产。



序号	名称	描述
PIN1	ANT GND	天线地
PIN2	ANT	天线驱动（50 Ohm antenna）
PIN3	VCC	模块电源 +5V
PIN4	VCC	模块电源 +5 V
PIN5	GND	电源地
PIN6	GND	电源地
PIN7	SDA/STA1	I2C 接口数据 I/O/UART 或 SPI 状态指示
PIN8	RXD/SCL	UART 接口的 RXD/I2C 接口的时钟信号
PIN9	TXD/STA2	UART 接口的 TXD/SPI 或 I2C 接口状态指示
PIN10	IDLE	休眠模式控制 0：工作 1： 休眠（休眠以后不接收任何命令）
PIN11	XTAL1	外接 13.56MHz 的晶体
PIN12	XTAL2	外接 13.56MHz 的晶体
PIN13	/SS	SPI 接口片选信号，输入口，低有效（BEEP）
PIN14	MOSI	SPI 接口数据输入
PIN15	RESET	模块内部 MCU 复位控制，低有效
PIN16	MISO/STA3	SPI 接口数据输出/UART 或 I2C 模式为模块状态指示
PIN17	SCK	SPI 接口时钟信号
PIN18	NC	保留未用
PIN19	NC	保留未用

复用管脚的说明：

- **PIN16（MISO/STA3）：**
SPI 接口下为数据输出脚。
UART 接口下，模块发送数据时 PIN16=0。模块空闲时 PIN16=1。
I2C 接口下，模块执行命令过程中 PIN16=0。模块空闲时 PIN16=1。
- **PIN7（SDA/STA1）：**
SPI 接口下，模块执行命令过程中 PIN7=0。模块空闲时 PIN7=1。
UART 接口下，模块执行命令过程中 PIN7=0。模块空闲时 PIN7=1。
I2C 接口下，数据输入/输出脚。
- **PIN8（RXD/SCL）：**
SPI 接口下，没有定义功能。
UART 接口下，数据接收端。
I2C 接口下，时钟输入端。
- **PIN19（TXD/STA2）：**
SPI 接口下，PIN19=0 表示模块准备好数据，等待 HOST 调取。发送完最后一个字节数据后复位为高电平。
UART 接口下，数据发送端。
I2C 接口下，PIN19=0 表示模块准备好数据，等待 HOST 调取。发送完最后一个字节数据后复位为高电平。

- **PIN13 (SS/BEEP) :**
SPI 接口下，设备选择端，输入状态，低有效。
UART 和 I2C 接口：BEEP 信号的输出，可以直接驱动 5V 蜂鸣器。
- **PIN11/PIN12 (XTAL1/XTAL2):**
可以直接连接 13.56MHz 的晶体振荡器。模块已经内置 20pF 电容和起振电路，无需其他外围电路。

附录 对 Mifare ONE 卡读写的实例

对第 4 个扇区进行操作

出厂新卡的 A、B 密钥均为 FF FF FF FF FF FF

以下命令是 对第 4 个扇区 A、B 密钥认证 的命令

```
A6 0B F4 1D 05 61 04 FF FF FF FF FF FF 00    B    key
A6 0B F4 1D 05 60 04 FF FF FF FF FF FF 01    A    key
```

以下命令是 对第 4 个扇区 读卡 的命令

A6 05 FA 1D 05 30 04 31

以下命令是 对第 4 个扇区的 写卡 的命令

A6 15 EA 1D 05 A0 04 22 22 22 22 22 22 22 22 22 22 22 22 22 9E

假设写入的内容为

0x22 0x22 0x22 0x22 0x22 0x22 0x22 0x22 0x22 0x22 0x22 0x22 0x22 0x22 0x22 0x22

对其它扇区的操作可参考前文提到的 **MIFARE ONE** 卡片操作说明