Unit RFID-UHF

常 用 控 制 指 令

目录

1、固件指令简介

1.1 指令帧格式

固件指令由帧头、帧类型、指令代码、指令数据长度、指令参数、校验码和帧尾组成，均为十六进制表示。例如：

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	指令参数	校验位	帧尾
BB	00	07	00	01	01	09	7E

帧头：　　0xBB
帧类型：　　0x00
指令代码：　　0x07
指令参数长度：　0x0001
指令参数：　　0x01
校验位：　　0x09
帧尾：　　0x7E

校验位为从帧类型到最后一个指令参数累加和，并只取累加和最低一个字节（LSB）。

1.2 指令帧类型

帧类型	描述
0x00	命令帧：由上位机发送给 M100 芯片
0x01	响应帧：由 M100 芯片发回给上位机
0x02	通知帧：由 M100 芯片发回给上位机

每一条指令帧都有对应的响应帧，响应帧表示指令是否已经被执行了。

单次轮询指令和多次轮询指令还有相应的通知帧，发送通知帧的个数是由 MCU 根据读取的情况，自主的发给上位机。当读写器读到一个标签就发一个通知帧，而当读写器读到多个标签时就会发送多个通知帧。

2、常用指令定义

2.1 获取读写器模块信息

2.1.1 命令帧

帧类型：　0x00
指令代码：　0x03
指令参数：　0x00-硬件版本　0x01-软件版本　0x02-制造商

- 硬件版本（00）：

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	指令参数	校验位	帧尾
BB	00	03	00	01	00	04	7E

- 软件版本（01）：

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	指令参数	校验位	帧尾
BB	00	03	00	01	01	05	7E

- 制造商（03）：

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	指令参数	校验位	帧尾
BB	00	03	00	01	02	06	7E

2.1.2 响应帧

帧类型： 0x01
指令代码： 0x03
指令参数： 0x00-硬件版本 0x01-软件版本 0x02-制造商
信息： ASCII 码

- 硬件版本：

例：M100 V1.00——ASCII：4D 31 30 30 20 56 31 2E 30 30

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	信息类型	版本信息	校验位	帧尾
BB	01	03	00	0B	00	见下表	22	7E

版本信息：

M	1	0	0		V	1	.	0	0
4D	31	30	30	20	56	31	2E	30	30

- 软件版本：

同理硬件版本信息。

- 制造商：

同理硬件版本信息。

2.2 单次轮询指令

2.2.1 命令帧

帧类型： 0x00
指令代码： 0x22
指令参数长度： 0x0000
校验位： 0x22

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	校验位	帧尾
BB	00	22	00	00	22	7E

2.2.2 通知帧

芯片接收到单次轮询指令后，如果能够读到 CRC 校验正确的标签，芯片 MCU 将返回包含 RSSI、PC、EPC 和 CRC 的数据。读到一个标签 EPC 就返回一条指令响应，读到多个标签则返回多条指令响应。如下：

帧类型： 0x02

指令代码： 0x22

指令参数长度： 0x0011

RSSI： 0xC9

PC： 0x3400

EPC： 0x30751FEB705C5904E3D50D70

CRC： 0x3A76

校验位： 0xEF

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	RSSI	PC（MSB）	PC（LSB）
BB	00	22	00	11	C9	34	00

EPC（MSB）										
30	75	1F	EB	70	5C	59	04	E3	D5	0D

EPC（LSB）	CRC（MSB）	CRC（LSB）	校验位	帧尾
70	3A	76	EF	7E

RSSI 值反映的是芯片输入端信号大小，不包含天线增益和定向耦合器衰减等。RSSI 为芯片输入端信号强度，十六进制有符号数，单位为 dBm。上面的例子中 RSSI 为 0xC9，代表芯片输入端信号强度为-55dBm。

2.2.3 响应帧

如果没有收到标签返回或者返回数据 CRC 校验错误，将返回错误代码 0x15，如下：

帧类型： 0x01

指令代码： 0xFF

指令参数长度： 0x01

指令参数： 0x15

校验位： 0x16

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	指令参数	校验位	帧尾
BB	01	FF	00	01	15	16	7E

2.3 多次轮询指令

2.3.1 命令帧

轮询次数限制为 0-65535 次，如果轮询次数为 10000 次，则指令如下：

帧类型： 0x00

指令代码： 0x27

指令参数长度： 0x0003

保留位： 0x22

轮询次数： 0x2710

校验位： 0x22

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	保留位	CNT（MSB）	CNT（LSB）	校验位	帧尾
BB	00	27	00	03	22	27	10	83	7E

2.3.2 通知帧

芯片接收到多次轮询指令后，如果能够读到 CRC 校验正确的标签，芯片 MCU 将返回包含 RSSI、PC、EPC 和 CRC 的数据。读到一个标签 EPC 就返回一条指令响应，读到多个标签则返回多条指令响应。如下：

帧类型： 0x02

指令代码： 0x27

指令参数长度： 0x0011

RSSI： 0xC9

PC： 0x3400

EPC： 0x30751FEB705C5904E3D50D70

CRC： 0x3A76

校验位： 0xEF

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	RSSI	PC（MSB）	PC（LSB）
BB	02	22	00	11	C9	34	00

EPC（MSB）										
30	75	1F	EB	70	5C	59	04	E3	D5	0D

EPC（LSB）	CRC（MSB）	CRC（LSB）	校验位	帧尾
70	3A	76	EF	7E

2.3.3 响应帧

如果没有收到标签返回或者返回数据 CRC 校验错误，将返回错误代码 0x15，如下：

帧类型： 0x01

指令代码： 0xFF

指令参数长度： 0x01

指令参数： 0x15

校验位： 0x16

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	指令参数	校验位	帧尾
BB	01	FF	00	01	15	16	7E

2.4 停止多次轮询指令

2.4.1 命令帧

立即停止多次轮询操作，非暂停多次轮询操作。

帧类型： 0x00

指令代码： 0x28

指令参数长度： 0x0000

校验位： 0x28

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	校验位	帧尾
BB	00	28	00	00	28	7E

2.4.2 响应帧

如果停止多次轮询指令成功执行，固件则返回响应如下：

帧类型： 0x01

指令代码： 0x28

指令参数长度： 0x0001

指令参数： 0x00

校验位： 0x2A

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	指令参数	校验位	帧尾
BB	01	28	00	01	00	2A	7E

2.5 设置 Select 参数指令

2.5.1 命令帧

设置 Select 参数，并且同时设置 Select 模式为 0x02(在对标签除轮询操作之前，先发送 Select 指令)。在多标签的情况下，可以根据 Select 参数只对特定标签进行轮询和读写等操作。例如：

帧类型： 0x00

指令代码： 0x0C

指令参数长度： 0x0013

SelParam： 0x01 (Target: 3’b000, Action: 3’b000, MemBank: 2’b01)

Ptr： 0x00000020 (以 bit 为单位，非 word) 从 EPC 存储位开始
Mask 长度： 0x60 (6 个 word，96bits)
是否 Truncate：0x00 (0x00 是 Disable truncation，0x80 是 Enable truncation)
Mask： 0x30751FEB705C5904E3D50D70
校验位： 0xAD

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	SelParam
BB	00	0C	00	13	01

Ptr（MSB）			Ptr（LSB）	MaskLen	Truncate
00	00	00	20	60	00

Mask（MSB）										
30	75	1F	EB	70	5C	59	04	E3	D5	0D

Mask（LSB）	校验位	帧尾
70	AD	7E

SelParam 共 1 个 Byte，其中 Target 占最高 3 个 bits，Action 占中间 3 个 bits，MemBank 占最后 2 个 bits。

MemBank 含义如下：

2’b00: 标签 RFU 数据存储区

2’b01: 标签 EPC 数据存储区

2’b10: 标签 TID 数据存储区

2’b11: 标签 User 数据存储区

Target 和 Action 详细含义请参见 EPC Gen2 协议。

当 Select Mask 长度大于 80 bits(5 words)，发送 Select 指令会先把场区内所有标签设置成 Inventoried Flag 为 A，SL Flag 为 ~SL 的状态，然后再根据所选的 Action 进行操作。当 Select Mask 长度小于 80 bits(5 words) 的时候，不会预先将标签状态通过 Select 指令设置成 Inventoried Flag 为 A，SL Flag 为 ~SL 的状态。

2.5.2 响应帧

当成功设置了 Select 参数后，固件返回如下：

帧类型： 0x01
指令代码： 0x0C
指令参数长度： 0x0001
返回数据： 0x00
校验位： 0x0E

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	返回数据	校验位	帧尾
BB	01	0C	00	01	00	0E	7E

2.6 获取 Select 参数

2.6.1 命令帧

帧类型： 0x00
指令代码： 0x0B

指令参数长度： 0x0000
校验位： 0x0B

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	校验位	帧尾
BB	00	0B	00	00	0B	7E

2.6.2 响应帧

帧类型： 0x01
指令代码： 0x0B
指令参数长度： 0x0013
SelParam： 0x01 (Target: 3’b000, Action: 3’b000, MemBank: 2’b01)
Ptr： 0x00000020 (以 bit 为单位，非 word) 从 EPC 存储位开始
Mask 长度： 0x60 (6 个 word，96bits)
是否 Truncate：0x00 (0x00 是 Disable truncation，0x80 是 Enable truncation)
Mask： 0x30751FEB705C5904E3D50D70
校验位： 0xAD

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	SelParam
BB	01	0B	00	13	01

Ptr（MSB）			Ptr（LSB）	MaskLen	Truncate
00	00	00	20	60	00

Mask（MSB）										
30	75	1F	EB	70	5C	59	04	E3	D5	0D

Mask（LSB）	校验位	帧尾
70	AD	7E

2.7 设置 Select 模式

2.7.1 命令帧

如果已经设置好了 Select 参数，执行该条指令，可以设置 Select 模式。例如，如果要取消 Select 指令：
帧类型： 0x00
指令代码： 0x12
指令参数长度： 0x0001
Select 模式： 0x01
校验位： 0x14

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	Select 模式	校验位	帧尾
BB	00	12	00	01	01	14	7E

Select 模式含义：
0x00：在对标签的所有操作之前都预先发送 Select 指令选取特定的标签。
0x01：在对标签操作之前不发送 Select 指令。
0x02：仅对除轮询 Inventory 之外的标签操作之前发送 Select 指令，如在

Read，Write，Lock，Kill 之前先通过 Select 选取特定的标签。

2.7.2 响应帧

当成功设置了取消或者发送 Select 指令后，固件返回如下：

帧类型： 0x01
指令代码： 0x0C
指令参数长度： 0x0001
返回数据： 0x00（执行成功）
校验位： 0x0E

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	指令参数	校验位	帧尾
BB	01	0C	00	01	00	0E	7E

2.8 读标签数据存储区

2.8.1 命令帧

对单个标签，读取标签数据存储区 Memory Bank 中指定地址和长度的数据。读标签数据区地址偏移 SA 和读取标签数据存储区长度 DL，他们的单位为 Word，即 2 个 Byte/16 个 Bit。这条指令之前应先设置 Select 参数，以便选择指定的标签进行读标签数据区操作。如果 Access Password 全为零，则不发送 Access 指令。

帧类型： 0x00
指令代码： 0x39
指令参数长度： 0x0009
Access Password： 0x0000FFFF
标签数据存储区 MemBank：0x03(User 区)
读标签数据区地址偏移 SA：0x0000 读标签
数据区地址长度 DL： 0x0002
校验位： 0x45

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	AP（MSB）		
BB	00	39	00	09	00	00	FF

AP（LSB）	MemBank	SA（MSB）	SA（LSB）	DL（MSB）	DL（LSB）	校验位	帧尾
FF	03	00	00	00	02	45	7E

2.8.2 响应帧

帧类型： 0x01
指令代码： 0x39
指令参数长度： 0x0013
操作的标签 PC+EPC 长度 UL：0x0E
操作标签 PC： 0x3400
操作标签 EPC： 0x30751FEB705C5904E3D50D70
返回数据： 0x12345678
校验位： 0xB0

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	UL	PC（MSB）	PC（LSB）
BB	01	39	00	13	0E	34	00

EPC（MSB）										
30	75	1F	EB	70	5C	59	04	E3	D5	0D

EPC（LSB）	返回数据（MSB）			返回数据（LSB）	校验位	帧尾
70	12	34	56	78	B0	7E

如果该标签没有在场区或者指定的 EPC 代码不对，会返回错误代码 0x09，如下：

帧类型： 0x01

指令代码： 0xFF

指令参数长度： 0x0001

指令参数： 0x09

校验位： 0x0A

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	指令参数	校验位	帧尾
BB	01	FF	00	01	09	0A	7E

如果 Access Password 不正确，则返回错误代码 0x16，并会返回所操作的标签的 PC+EPC，如下：

帧类型： 0x01

指令代码： 0xFF

指令参数长度： 0x0010

指令参数： 0x16

PC+EPC 长度 UL：0x0E

PC： 0x3400

EPC： 0x30751FEB705C5904E3D50D70

校验位： 0x75

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	指令参数	UL	PC（MSB）	PC（LSB）
BB	01	FF	00	10	16	0E	34	00

EPC（MSB）										
30	75	1F	EB	70	5C	59	04	E3	D5	0D

EPC（LSB）	校验位	帧尾
70	75	7E

如果操作标签返回了 EPC Gen2 协议规定的错误代码(error codes)，因为 EPC Gen2 规定的错误代码只有低 4 位有效，响应帧会将标签返回的错误代码或上 0xA0 之后再返回。比如如果发送指令参数中地址偏移或者数据长度不正确，读取数据长度超过标签数据存储区长度，按照 EPC Gen2 协议，标签会返回错误代码 0x03(存储区超出，Memory Overrun)。响应帧则返回错误代码 0xA3，并返回所操作标签的 PC+EPC，如下：

帧类型： 0x01

指令代码： 0xFF

指令参数长度： 0x0010

指令参数： 0xA3

PC+EPC 长度 UL：0x0E

PC： 0x3400
EPC： 0x30751FEB705C5904E3D50D70
校验位： 0x02

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	错误代码	UL	PC（MSB）	PC（LSB）
BB	01	FF	00	10	A3	0E	34	00

EPC（MSB）										
30	75	1F	EB	70	5C	59	04	E3	D5	0D

EPC（LSB）	校验位	帧尾
70	02	7E

2.9 写标签数据存储区

2.9.1 命令帧

对单个标签，写入标签数据存储区 Memory Bank 中指定地址和长度的数据。标签数据区地址偏移 SA 和要写入的标签数据长度 DL，他们的单位为 Word，即 2 个 Byte/16 个 Bit。这条指令之前应先设置 Select 参数，以便选择指定的标签进行写标签数据区操作。如果 Access Password 全为零，则不发送 Access 指令。

写入标签数据存储区的数据长度 DT 应不超过 32 个 word，即 64 字节/512 位。

帧类型： 0x00
指令代码： 0x49
指令参数长度： 0x000D
Access Password： 0x0000FFFF
标签数据存储区 MemBank：0x03
读标签数据区地址偏移 SA：0x0000 数据
数据长度 DL： 0x0002 写入
数据 DT： 0x12345678
校验位： 0x6D

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	AP（MSB）		
BB	00	49	00	0D	00	00	FF

AP（LSB）	MemBank	SA（MSB）	SA（LSB）	DL（MSB）	DL（LSB）
FF	03	00	00	00	02

DT（MSB）			DT（LSB）	校验位	帧尾
12	34	56	78	6D	7E

2.9.2 响应帧

将数据写入标签数据存储区后，如果读写器芯片接收到标签返回值正确，则响应帧如下：

帧类型： 0x01
指令代码： 0x39
指令参数长度： 0x0010
操作的标签 PC+EPC 长度 UL：0x0E
操作标签 PC： 0x3400

操作标签 EPC：　　0x30751FEB705C5904E3D50D70
指令参数：　　0x00（执行成功）
校验位：　　0xA9

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	UL	PC（MSB）	PC（LSB）
BB	01	49	00	10	0E	34	00

EPC（MSB）									
30	75	1F	EB	70	5C	59	04	E3	D5	0D

EPC（LSB）	指令参数	校验位	帧尾
70	00	A9	7E

如果该标签没有在场区或者指定的 EPC 代码不对，会返回错误代码 0x10，如下：
帧类型：　　0x01
指令代码：　　0xFF
指令参数长度：　　0x0001
指令参数：　　0x10
校验和：　　0x0A

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	指令参数	校验位	帧尾
BB	01	FF	00	01	10	0A	7E

如果 Access Password 不正确，则返回错误代码 0x16，并会返回所操作的标签的 PC+EPC，如下：
帧类型：　　0x01
指令代码：　　0xFF
指令参数长度：　　0x0016
指令参数：　　0x16
PC+EPC 长度 UL：0x0E
PC：　　0x3400
EPC：　　0x30751FEB705C5904E3D50D70
校验位：　　0x75

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	指令参数	UL	PC（MSB）	PC（LSB）
BB	01	FF	00	10	16	0E	34	00

EPC（MSB）									
30	75	1F	EB	70	5C	59	04	E3	D5	0D

EPC（LSB）	校验位	帧尾
70	75	7E

如果操作标签返回了 EPC Gen2 协议规定的错误代码(error codes)，因为 EPC Gen2 规定的错误代码只有低 4 位有效，响应帧会将标签返回的错误代码或上 0xB0 之后再返回。比如如果发送指令参数中地址偏移或者数据长度不正确，读取数据长度超过标签数据存储区长度，按照 EPC Gen2 协议，标签会返回错误代码 0x03(存储区超出，Memory Overrun)。响应帧则返回错误代码 0xB3，并返回所操作标签的 PC+EPC，如下：
帧类型：　　0x01
指令代码：　　0xFF

指令参数长度： 0x0010
指令参数： 0xB3
PC+EPC 长度 UL：0x0E
PC： 0x3400
EPC： 0x30751FEB705C5904E3D50D70
校验位： 0x12

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	错误代码	UL	PC（MSB）	PC（LSB）
BB	01	FF	00	10	B3	0E	34	00

EPC（MSB）										
30	75	1F	EB	70	5C	59	04	E3	D5	0D

EPC（LSB）	校验位	帧尾
70	12	7E

2.10 锁定 Lock 标签数据存储区

2.10.1 命令帧

对单个标签，锁定 Lock 或者解锁 Unlock 该标签的数据存储区。这条指令之前应先设置 Select 参数，以便选择指定的标签进行锁定 Lock 操作。例如要锁定 Access Password，则指令如下：

帧类型： 0x00
指令代码： 0x82
指令参数长度： 0x0007
Access Password： 0x0000FFFF
Lock 操作数 LD： 0x020080
校验位： 0x09

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	AP（MSB）		
BB	00	82	00	07	00	00	FF

AP（LSB）	LD（MSB）		LD（LSB）	校验位	帧尾
FF	02	00	80	09	7E

Lock 操作参数 LD 的高 4 位是保留位，剩下的 20 为是 Lock 操作 Payload，包括 Mask 和 Action，从高到低依次各 10 位。详细含义请参见 EPC Gen2 协议 1.2.0 版 6.3.2.11.3.5 节。

Mask 是一个掩膜，只有 Mask 位为 1 的 Action 才有效。每个数据区的 Action 有 2 bits，00~11，依次对应为开放，永久开放，锁定，永久锁定。

比如 Kill Mask 为 2bits 00，则不管 Kill Action 是什么，Kill Action 都不会生效。当 Kill Mask 为 2bits 10，Kill Action 为 2bits 10，代表 Kill Password 被 Lock(非 Perma Lock)住了，只有通过有效的 Access Password 才能被读写。

Mask 和 Action 每一位的含义如下表示。

Lock-Command Payload

19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Kill Mask		Access Mask		EPC Mask		TID Mask		User Mask		Kill Action		Access Action		EPC Action		TID Action		User Action	

Masks and Associated Action Fields

	Kill pwd		Access pwd		EPC memory		TID memory		User memory	
	19	18	17	16	15	14	13	12	11	10
Mask	skip/ write	skip/ write	skip/ write	skip/ write	skip/ write	skip/ write	skip/ write	skip/ write	skip/ write	skip/ write
	9	8	7	6	5	4	3	2	1	0
Action	pwd read/ write	perma lock	pwd read/ write	perma lock	pwd write	perma lock	pwd write	perma lock	pwd write	perma lock

pwd-write	permalock	Description
0	0	Associated memory bank is writeable from either the **open** or **secured** states.
0	1	Associated memory bank is permanently writeable from either the **open** or **secured** states and may never be locked.
1	0	Associated memory bank is writeable from the **secured** state but not from the **open** state.
1	1	Associated memory bank is not writeable from any state.
pwd-read/write	**permalock**	**Description**
0	0	Associated password location is readable and writeable from either the **open** or **secured** states.
0	1	Associated password location is permanently readable and writeable from either the **open** or **secured** states and may never be locked.
1	0	Associated password location is readable and writeable from the **secured** state but not from the **open** state.
1	1	Associated password location is not readable or writeable from any state.

2.10.2 响应帧

如果 Lock 指令执行正确，标签的返回有效，则响应帧为：

帧类型： 0x01
指令代码： 0x82
指令参数长度： 0x0010
操作的标签 PC+EPC 长度 UL：0x0E
操作标签 PC： 0x3400
操作标签 EPC： 0x30751FEB705C5904E3D50D70
返回数据： 0x00（执行成功）
校验位： 0xE2

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	UL	PC（MSB）	PC（LSB）
BB	01	82	00	10	0E	34	00

EPC（MSB）										
30	75	1F	EB	70	5C	59	04	E3	D5	0D

EPC（LSB）	指令参数	校验位	帧尾
70	00	E2	7E

如果该标签没有在场区或者指定的 EPC 代码不对，会返回错误代码 0x13，如下：

帧类型： 0x01
指令代码： 0xFF
指令参数长度： 0x0001

指令参数： 0x13
校验位： 0x14

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	指令参数	校验位	帧尾
BB	01	FF	00	01	13	14	7E

如果 Access Password 不正确，则返回错误代码 0x16，并会返回所操作的标签的 PC+EPC，如下：

帧类型： 0x01
指令代码： 0xFF
指令参数长度： 0x0016
指令参数： 0x16
PC+EPC 长度 UL：0x0E
PC： 0x3400
EPC： 0x30751FEB705C5904E3D50D70
校验位： 0x75

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	指令参数	UL	PC（MSB）	PC（LSB）
BB	01	FF	00	10	16	0E	34	00
EPC（MSB）								
30	75	1F	EB	70	5C	59	04	E3	D5	0D
EPC（LSB）				校验位			帧尾	
70				75			7E	

如果操作标签返回了 EPC Gen2 协议规定的错误代码(error codes)，因为 EPC Gen2 规定的错误代码只有低 4 位有效，响应帧会将标签返回的错误代码或上 0xC0 之后再返回。比如如果发送指令参数中地址偏移或者数据长度不正确，读取数据长度超过标签数据存储区长度，按照 EPC Gen2 协议，标签会返回错误代码 0x04(存储区超出，Memory Overrun)。响应帧则返回错误代码 0xC4，并返回所操作标签的 PC+EPC，如下：

帧类型： 0x01
指令代码： 0xFF
指令参数长度： 0x0010
指令参数： 0xC4
PC+EPC 长度 UL：0x0E
PC： 0x3400
EPC： 0x30751FEB705C5904E3D50D70
校验位： 0x23

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	错误代码	UL	PC（MSB）	PC（LSB）
BB	01	FF	00	10	C4	0E	34	00
EPC（MSB）								
30	75	1F	EB	70	5C	59	04	E3	D5	0D
EPC（LSB）				校验位			帧尾	
70				23			7E	

2.11 灭活 Kill 标签

2.11.1 命令帧

这条指令之前应先设置 Select 参数，以便选择指定的标签进行灭活 Kill 操作。对单标签的灭活操作。

帧类型： 0x00
指令代码： 0x65
指令参数长度： 0x0004
Kill Password： 0x0000FFFF
校验位： 0x67

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	KP（MSB）		
BB	00	65	00	04	00	00	FF

KP（LSB）	校验位	帧尾
70	67	7E

2.11.2 响应帧

如果 Kill 指令执行正确，标签的返回 CRC 正确，则响应帧为：

帧类型： 0x00
指令代码： 0x65
指令参数长度： 0x0010
操作的标签 PC+EPC 长度 UL：0x0E
操作标签 PC： 0x3400
操作标签 EPC： 0x30751FEB705C5904E3D50D70
返回数据： 0x00（执行成功）
校验位： 0xC5

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	UL	PC（MSB）	PC（LSB）
BB	01	65	00	10	0E	34	00

EPC（MSB）										
30	75	1F	EB	70	5C	59	04	E3	D5	0D

EPC（LSB）	指令参数	校验位	帧尾
70	00	C5	7E

如果该标签没有在场区或者指定的 EPC 代码不对，会返回错误代码 0x12，如下：

帧类型： 0x01
指令代码： 0xFF
指令参数长度： 0x0001
指令参数： 0x12
校验位： 0x13

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	指令参数	校验位	帧尾
BB	01	FF	00	01	12	13	7E

如果操作标签返回了 EPC Gen2 协议规定的错误代码(error codes)，响应帧会将标签返回的错误代码或上 0xD0 之后再返回。

注意，标签如果没有设置过 Kill Password 密码，即 Kill Password 密码全为 0，按照 Gen2 协议，标签不会被 Kill。这时返回错误代码 0xD0，如下：

帧类型： 0x01
指令代码： 0xFF
指令参数长度： 0x0010
指令参数： 0xD0
PC+EPC 长度 UL：0x0E
PC： 0x3400
EPC： 0x30751FEB705C5904E3D50D70
校验位： 0x2F

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	指令参数	UL	PC（MSB）	PC（LSB）
BB	01	FF	00	10	D0	0E	34	00

EPC（MSB）										
30	75	1F	EB	70	5C	59	04	E3	D5	0D

EPC（LSB）	校验位	帧尾
70	2F	7E

2.12 设置通信波特率

2.12.1 命令帧

连接上读写器后，设置之后的通信波特率，比如设置成 19200 命令帧定义如下：

帧类型： 0x00
指令代码： 0x11
指令参数长度：0x0002
功率参数 Pow：0x00C0 (波特率/100 的 16 进制，比如 19200，就是 19200/100=192=0xC0)
校验位： 0x45

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	Pow（MSB）	Pow（LSB）	校验位	帧尾
BB	00	11	00	02	00	C0	D3	7E

2.12.2 响应帧

该指令没有响应帧。读写器执行完设置通信波特率指令后，读写器就会用新的波特率与上位机通信，上位机需 要用新的波特率重新连接读写器。

2.13 获取 Query 参数

2.13.1 命令帧

帧类型： 0x00
指令代码： 0x0D

指令参数长度：　　0x0000
校验位：　　0x0D

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	校验位	帧尾
BB	00	0D	00	00	0D	7E

2.13.2 响应帧

帧类型：　　0x01
指令代码：　　0x0D
指令参数长度：　　0x0002
Query Parameter：　0x1020
校验位：　　0x40

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	Para（MSB）	Para（LSB）	校验位	帧尾
BB	01	0D	00	02	10	20	40	7E

参数为 2 字节，有下面的具体参数按位拼接而成。上述响应帧对应的 Query 参数为：
DR=8，M=1，TRext=Use pilot tone，Sel=00，Session=00，Target=A，Q=4
其中：
DR(1 bit)：DR=8(1’b0)，DR=64/3(1’b1)。只支持 DR=8 的模式
M(2 bit)：M=1(2’b00)，M=2(2’b01)，M=4(2’b10)，M=8(2’b11). 只支持 M=1 的模式
TRext(1 bit)：No pilot tone(1’b0)，Use pilot tone(1’b1)。只支持 Use pilot tone(1’b1)模式
Sel(2 bit)：ALL(2’b00/2’b01), ~SL(2’b10), SL(2’b11)
Session(2 bit)：S0(2’b00), S1(2’b01), S2(2’b10), S3(2’b11)
Target(1 bit)：A(1’b0), B(1’b1)
Q(4 bit)：4’b0000-4’b1111

2.14 设置 Query 参数

2.14.1 命令帧

设置 Query 命令中的相关参数。参数为 2 字节，有下面的具体参数按位拼接而成：
DR(1 bit)：DR=8(1’b0)，DR=64/3(1’b1)。只支持 DR=8 的模式
M(2 bit)：M=1(2’b00)，M=2(2’b01)，M=4(2’b10)，M=8(2’b11). 只支持 M=1 的模式
TRext(1 bit)：No pilot tone(1’b0)，Use pilot tone(1’b1)。只支持 Use pilot tone(1’b1)模式
Sel(2 bit)：ALL(2’b00/2’b01), ~SL(2’b10), SL(2’b11)
Session(2 bit)：S0(2’b00), S1(2’b01), S2(2’b10), S3(2’b11)
Target(1 bit)：A(1’b0), B(1’b1)
Q(4 bit)：4’b0000-4’b1111
如果 DR=8，M=1，TRext=Use pilot tone，Sel=00，Session=00，Target=A，Q=4，则指令如下：

帧类型：　　0x00
指令代码：　　0x0E
指令参数长度：　　0x0002
Query Parameter：　0x1020

校验位： 0xC6

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	Para（MSB）	Para（LSB）	校验位	帧尾
BB	00	0E	00	02	10	20	40	7E

2.14.2 响应帧

如果设置 Query 参数指令执行正确，则响应帧为：

帧类型： 0x01
指令代码： 0x0E
指令参数长度： 0x0001
指令参数： 0x00
校验位： 0x10

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	指令参数	校验位	帧尾
BB	01	0E	00	01	00	10	7E

2.15 设置工作地区

2.15.1 命令帧

设置读写器工作地区，如果是中国 900MHz 频段，如下：

帧类型： 0x00
指令代码： 0x07
指令参数长度： 0x0001
地区： 0x01
校验位： 0x09

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	地区	校验位	帧尾
BB	00	07	00	01	01	09	7E

不同国家地区代码如下表：

地区	代码
中国 900MHz	01
中国 800MHz	04
美国	02
欧洲	03
韩国	06

2.15.2 响应帧

如果地区设置执行正确，则响应帧为：

帧类型： 0x01
指令代码： 0x07
指令参数长度： 0x0001
指令参数： 0x00

校验位： 0x09

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	指令参数	校验位	帧尾
BB	01	07	00	01	00	09	7E

2.16 获取工作地区

2.16.1 命令帧

帧类型： 0x00
指令代码： 0x08
指令参数长度： 0x0000
校验位： 0x08

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	校验位	帧尾
BB	00	08	00	00	08	7E

2.16.2 响应帧

设置读写器工作地区，如果是中国 900MHz 频段，如下：

帧类型： 0x01
指令代码： 0x08
指令参数长度： 0x0001
地区： 0x01
校验位： 0x0B

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	地区	校验位	帧尾
BB	01	08	00	01	01	0B	7E

不同国家地区代码如下表：

地区	代码
中国 900MHz	01
中国 800MHz	04
美国	02
欧洲	03
韩国	06

2.17 设置工作信道

2.17.1 命令帧

如果是中国 900MHz 频段，设置读写器工作信道 920.375MHz，如下：

帧类型： 0x00
指令代码： 0xAB
指令参数长度： 0x0001

地区： 0x01
校验位： 0xAD

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	地区	校验位	帧尾
BB	00	AB	00	01	01	AD	7E

中国 900MHz 信道参数计算公式，Freq_CH 为信道频率：

CH_Index = (Freq_CH-920.125M)/0.25M

中国 800MHz 信道参数计算公式，Freq_CH 为信道频率：

CH_Index = (Freq_CH-840.125M)/0.25M

美国信道参数计算公式，Freq_CH 为信道频率：

CH_Index = (Freq_CH-902.25M)/0.5M

欧洲信道参数计算公式，Freq_CH 为信道频率：

CH_Index = (Freq_CH-865.1M)/0.2M

韩国信道参数计算公式，Freq_CH 为信道频率：

CH_Index = (Freq_CH-917.1M)/0.2M

2.17.2 响应帧

如果信道设置执行正确，则响应帧为：

帧类型： 0x01
指令代码： 0xAB
指令参数长度： 0x0001
指令参数： 0x00
校验位： 0xAD

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	指令参数	校验位	帧尾
BB	01	AB	00	01	00	AD	7E

2.18 获取工作信道

2.18.1 命令帧

在当前的读写器工作地区，获取读写器工作信道，如下：

帧类型： 0x00
指令代码： 0xAA
指令参数长度： 0x0000
校验位： 0xAA

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	校验位	帧尾
BB	00	AA	00	00	AA	7E

2.18.2 响应帧

如果获取信道执行正确，则命令帧响应为：

帧类型： 0x01
指令代码： 0xAA
指令参数长度： 0x0001
指令参数： 0x00
校验位： 0xAC

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	指令参数	校验位	帧尾
BB	01	AA	00	01	00	AC	7E

中国 900MHz 信道参数计算公式，Freq_CH 为信道频率：

Freq_CH = CH_Index * 0.25M + 920.125M

中国 800MHz 信道参数计算公式，Freq_CH 为信道频率：

Freq_CH = CH_Index * 0.25M + 840.125M

美国信道参数计算公式，Freq_CH 为信道频率：

Freq_CH = CH_Index * 0.5M + 902.25M

欧洲信道参数计算公式，Freq_CH 为信道频率：

Freq_CH = CH_Index * 0.2M + 865.1M

韩国信道参数计算公式，Freq_CH 为信道频率：

Freq_CH = CH_Index * 0.2M + 917.1M

2.19 设置自动跳频

2.19.1 命令帧

设置为自动跳频模式或者取消自动跳频模式。在自动跳频模式下，如果用户执行了插入工作信道指令，则读写 器从用户设置的信道列表中随机选择信道跳频，否则按照内部预设的信道列表随机选择信道跳频。指令格式如下：

帧类型： 0x00
指令代码： 0xAD
指令参数长度： 0x0001
指令参数： 0xFF (0xFF 为设置自动跳频，0x00 为取消自动跳频)
校验位： 0xAD

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	指令参数	校验位	帧尾
BB	00	AD	00	01	FF	AD	7E

2.19.2 响应帧

如果设置为自动跳频行或者取消自动跳频正确，则响应帧为：

帧类型： 0x00
指令代码： 0xAD

指令参数长度： 0x0001
指令参数： 0x00
校验位： 0xAF

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	指令参数	校验位	帧尾
BB	01	AD	00	01	00	AF	7E

2.20 插入工作信道

2.20.1 命令帧

插入工作信道可以让用户自主设置跳频的信道列表，执行此命令后，读写器将从用户设置的信道列表中随机选 择信道跳频，命令定义如下：

帧类型： 0x00
指令代码： 0xA9
指令参数长度： 0x0006
信道个数： 0x05
信道列表： 0x01 0x02 0x03 0x04 0x05
校验位： 0xC3

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	信道个数
BB	00	A9	00	06	05

信道列表（MSB）				信道列表（LSB）	校验位	帧尾
01	02	03	04	05	C3	7E

2.20.2 响应帧

如果执行正确，则响应帧为：

帧类型： 0x01
指令代码： 0xA9
指令参数长度： 0x0001
指令参数： 0x00
校验位： 0xAB

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	指令参数	校验位	帧尾
BB	01	A9	00	01	00	AB	7E

2.21 获取发射功率

2.21.1 命令帧

帧类型： 0x00
指令代码： 0xB7
指令参数长度： 0x0000
校验位： 0xB7

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	校验位	帧尾
BB	00	B7	00	00	B7	7E

2.21.2 响应帧

如果执行正确，则响应帧为：

帧类型： 0x01
指令代码： 0xB7
指令参数长度： 0x0002
功率参数 Pow： 0x07D0 (当前功率为十进制 2000，即 20dBm)
校验位： 0x91

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	Pow（MSB）	Pow（LSB）	校验位	帧尾
BB	01	B7	00	02	07	D0	91	7E

2.22 设置发射功率

2.22.1 命令帧

帧类型： 0x00
指令代码： 0xB6
指令参数长度： 0x0002
功率参数 Pow： 0x07D0 (当前功率为十进制 2000，即 20dBm)
校验位： 0x8F

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	Pow（MSB）	Pow（LSB）	校验位	帧尾
BB	00	B6	00	02	07	D0	8F	7E

2.22.2 响应帧

如果获取信道执行正确，则响应帧为：

帧类型： 0x01
指令代码： 0xB6
指令参数长度： 0x0001
指令参数： 0x00
校验位： 0xB8

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	指令参数	校验位	帧尾
BB	01	B6	00	01	00	B8	7E

2.23 设置发射连续载波

2.23.1 命令帧

设置发射连续载波或者关闭连续载波，如下：

帧类型： 0x00
指令代码： 0xB0
指令参数长度： 0x0001
指令参数： 0xFF (0xFF 为打开连续波，0x00 为关闭连续波)
校验位： 0xB0

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	指令参数	校验位	帧尾
BB	00	B0	00	01	FF	B0	7E

2.23.2 响应帧

如果设置执行正确，则响应帧为：

帧类型： 0x01
指令代码： 0xB0
指令参数长度： 0x0001
指令参数： 0x00
校验位： 0xB2

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	指令参数	校验位	帧尾
BB	01	B0	00	01	00	B2	7E

2.24 获取接收解调器参数

2.24.1 命令帧

获取当前读写器接收解调器参数。解调器参数有 Mixer 增益，中频放大器 IF AMP 增益和信号解调阈值。例如：

帧类型： 0x00
指令代码： 0xF1
指令参数长度： 0x0000
校验位： 0xF1

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	校验位	帧尾
BB	00	F1	00	00	F1	7E

2.24.2 响应帧

帧类型： 0x01
指令代码： 0xF1
指令参数长度： 0x0004

混频器增益 Mixer_G： 0x03 (混频器 Mixer 增益为 9dB)
中频放大器增益 IF_G：0x06 (中频放大器 IF AMP 增益为 36dB)
信号解调阈值 Thrd： 0x01B0 (信号解调阈值越小能解调的标签返回 RSSI 越低，但越不稳定，低于一定值完全不能解调；相反阈值越大能解调的标签返回信号 RSSI 越大，距离越近，越稳定。0x01B0 是推荐的最小值)
校验位： 0xB0

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	Mixer_G	IF_G	Thrd（MSB）	Thrd（LSB）	校验位	帧尾
BB	01	F1	00	04	03	06	01	B0	B0	7E

混频器 **Mixer** 增益表

帧类型	Mixer_G(dB)
0x00	0
0x01	3
0x02	6
0x03	9
0x04	12
0x05	15
0x06	16

中频放大器 **IF AMP** 增益表

帧类型	IF_G(dB)
0x00	12
0x01	18
0x02	21
0x03	24
0x04	27
0x05	30
0x06	36
0x07	40

2.25 设置接收解调器参数

2.25.1 命令帧

设置当前读写器接收解调器参数。解调器参数有 Mixer 增益，中频放大器 IF AMP 增益和信号解调阈值。例如：

帧类型： 0x00
指令代码： 0xF0
指令参数长度： 0x0004
混频器增益 Mixer_G： 0x03 (混频器 Mixer 增益为 9dB)
中频放大器增益 IF_G：0x06 (中频放大器 IF AMP 增益为 36dB)
信号解调阈值 Thrd： 0x01B0 (信号解调阈值越小能解调的标签返回 RSSI 越低，但越不稳定，低于一定值完全不能解调；相反阈值越大能解调的标签返回信号 RSSI 越大，距离越近，越稳定。0x01B0 是推荐的最小值)

校验位： 0xAE

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	Mixer_G	IF_G	Thrd（MSB）	Thrd（LSB）	校验位	帧尾
BB	00	F0	00	04	03	06	01	B0	AE	7E

混频器 **Mixer** 增益表

帧类型	Mixer_G(dB)
0x00	0
0x01	3
0x02	6
0x03	9
0x04	12
0x05	15
0x06	16

中频放大器 **IF AMP** 增益表

帧类型	IF_G(dB)
0x00	12
0x01	18
0x02	21
0x03	24
0x04	27
0x05	30
0x06	36
0x07	40

2.25.2 响应帧

如果获取信道执行正确，则响应帧为：

帧类型： 0x01
指令代码： 0xF0
指令参数长度： 0x0001
指令参数： 0x00
校验位： 0xF2

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	指令参数	校验位	帧尾
BB	01	F0	00	01	00	F2	7E

2.26 测试射频输入端阻塞信号

2.26.1 命令帧

测试射频输入端阻塞信号 Scan Jammer，用于检测读写器天线在当前地区每个信道的阻塞信号大小。例如：

帧类型： 0x00
指令代码： 0xF2

指令参数长度： 0x0000
校验位： 0xF2

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	校验位	帧尾
BB	00	F2	00	00	F2	7E

2.26.2 响应帧

如果在中国 900MHz 频段下，一共 20 个信道，测试射频输入端阻塞信号 Scan Jammer 道执行正确，则响应帧为：

帧类型： 0x01
指令代码： 0xF2
指令参数长度： 0x0016
测试起始信道 CH_L：0x00 (测试起始信道 Index 为 0)
测试结束信道 CH_H：0x13 (测试起始信道 Index 为 19)
信道阻塞信号 JMR： 0xF2F1F0EFECEAE8EAECEEF0F1F5F5F5F6F5F5F5F5 (每个信道的阻塞信号 JMR 都用一个有符号的 Byte 表示，其中 0xF2 为-14dBm)
校验位： 0xDD

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	CH_L	CH_H
BB	01	F2	00	16	00	13

JMR（MSB）																	
F2	F1	F0	EF	EC	EA	E8	EA	EC	EE	F0	F1	F5	F5	F5	F6	F5	F5	F5

JMR（LSB）	校验位	帧尾
F5	DD	7E

2.27 测试信道 RSSI

2.27.1 命令帧

测试射频输入端 RSSI 信号大小，用于检测当前环境下有无读写器在工作。例如：

帧类型： 0x00
指令代码： 0xF3
指令参数长度： 0x0000
校验位： 0xF3

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	校验位	帧尾
BB	00	F3	00	00	F3	7E

2.27.2 响应帧

如果在中国 900MHz 频段下，一共 20 个信道，检测每个信道 RSSI 道执行正确，则响应帧为：

帧类型： 0x01
指令代码： 0xF3

指令参数长度： 0x0016
测试起始信道 CH_L：0x00 (测试起始信道 Index 为 0)
测试结束信道 CH_H：0x13 (测试起始信道 Index 为 19)
信道阻塞信号 JMR： 0xBABABABABABABABABABABABABABABABABABABABA (每个信道的 RSSI 用一个有符号的 Byte 表示，其中 0xBA 为-70dBm，为读写器能检测的 RSSI 的最小值)
校验位： 0xA5

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	CH_L	CH_H
BB	01	F3	00	16	00	13

RSSI（MSB）																			
BA	BA	BA	BA	BA	BA	BA	BA	BA	BA	BA	BA	BA	BA	BA	BA	BA	BA	BA	BA

RSSI（LSB）	校验位	帧尾
BA	A5	7E

2.28 控制 IO 端口

2.28.1 命令帧

设置 IO 端口的方向，读取 IO 电平以及设置 IO 电平，如下：

帧类型： 0x00
指令代码： 0x1A
指令参数长度： 0x0003
指令参数： 0x00 0x04 0x01
校验位： 0x22

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	指令参数			校验位	帧尾
BB	01	1A	00	03	00	04	01	22	7E

参数说明：

编号	描述	长度	说明
0	参数 0	1 Byte	操作类型选择： 0x00：设置 IO 方向 0x01：设置 IO 电平 0x02：读取 IO 电平，要操作的管脚在参数 1 中指定。
1	参数 1	1 Byte	参数值范围为 0x01~0x04，分别对应要操作的端口 IO1~IO4
2	参数 2	1 Byte	参数值为 0x00 或 0x01 参数 0 | 参数 2 | 描述 0x00 | 0x00 | IO 配置为输入模式 0x00 | 0x01 | IO 配置为输出模式 0x01 | 0x00 | 设置 IO 输出为低电平 0x01 | 0x01 | 设置 IO 输出为高电平 当参数 0 为 0x02 时，此参数无意义。

2.28.2 响应帧

帧类型： 0x01
指令代码： 0x1A
指令参数长度： 0x0003
指令参数： 0x00 0x04 0x01
校验位： 0x23

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	指令参数			校验位	帧尾
BB	01	1A	00	03	00	04	01	23	7E

参数说明：

编号	描述	长度	说明
0	参数 0	1 Byte	操作类型选择： 0x00：设置 IO 方向 0x01：设置 IO 电平 0x02：读取 IO 电平，要操作的管脚在参数 1 中指定。
1	参数 1	1 Byte	参数值范围为 0x01~0x04，分别对应要操作的端口 IO1~IO4
2	参数 2	1 Byte	参数值为 0x00 或 0x01 参数 0 | 参数 2 | 描述 0x00 | 0x00 | 表示 IO 配置失败 0x00 | 0x01 | 表示 IO 配置成功 0x01 | 0x00 | 表示设置 IO 输出失败 0x01 | 0x01 | 表示设置 IO 输出成功 0x02 | 0x00 | 表示对应端口为低电平 0x02 | 0x01 | 表示对应端口为高电平

2.29 模块休眠

2.29.1 命令帧

模块休眠指令可以让模块保持低功耗的休眠模式。模块休眠后，通过串口发送任意的字节即可唤醒模块，但该字节会被抛弃掉，模块休眠后接收到的第一条指令会没有响应，因为第一条指令的第一个字符会被抛弃掉。该指令会让 M100/QM100 芯片掉电重置，模块唤醒后会立刻重新下载固件到 M100/QM100 芯片中，并重新设置一些参数到模块中（这些参数包括休眠前配置的功率，频率，跳频模式，休眠时间，接收解调器参数，不包括 Select 模式，Select 参数等），因此有些参数可能需要重新设置。指令如下：

帧类型： 0x00
指令代码： 0x17
指令参数长度： 0x0000
校验位： 0x17

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	校验位	帧尾
BB	00	17	00	00	17	7E

2.29.2 响应帧

如果执行成功，则响应帧为：

帧类型： 0x01
指令代码： 0x17
指令参数长度： 0x0001
指令参数： 0x00
校验位： 0x19

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	指令参数	校验位	帧尾
BB	01	17	00	01	00	19	7E

2.30 模块空闲休眠时间

2.30.1 命令帧

该指令可以设置模块经过多长时间没有操作后自动进入休眠状态。模块休眠后，通过串口发送任意的字符即可 唤醒模块。模块休眠后接收到的第一条指令会没有响应，因为第一条指令的第一个字符会被抛弃掉。该指令会 让 M100/QM100 芯片重置，模块唤醒后会立刻重新下载固件到 M100/QM100 芯片中，并重新设置一些参数 到模块中（这些参数包括休眠前配置的功率，频率，跳频模式，休眠时间，接收解调器参数，不包括 Select 模式，Select 参数等），因此有些参数可能需要重新设置。指令如下：

帧类型： 0x00
指令代码： 0x1D
指令参数长度： 0x0001
指令参数： 0x02 (2 分钟无操作之后休眠，范围 1~30 分钟，0x00 代表不自动休眠)
校验位： 0x17

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	指令参数	校验位	帧尾
BB	00	1D	00	01	02	20	7E

2.30.2 响应帧

如果执行成功，则响应帧为：

帧类型： 0x01
指令代码： 0x1D
指令参数长度： 0x0001
指令参数： 0x02
校验位： 0x21

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	指令参数	校验位	帧尾
BB	01	1D	00	01	02	21	7E

2.31 IDLE 模式

2.31.1 命令帧

该指令可以让模块进入 IDLE 工作模式，IDLE 模式下除了数字部分和通信接口，其余所有模拟和射频部分电源均被关闭，以减少不工作情况下的功耗。模块进入 IDLE 模式后，与模块仍可正常通信，已设置的参数仍然被保存，模块可以正常响应上位机的指令。进入 IDLE 模式后，第一次盘点（或读取，写入标签数据此类需要与标签交互的指令）会让模块恢复到正常状态，但第一次盘点可能由于射频部分电源状态不稳定导致成功率下降，后续的盘点和其他操作即可恢复正常。指令如下：

帧类型： 0x00
指令代码： 0x04
指令参数长度： 0x0003
是否进入 IDLE 模式 Enter： 0x01 (进入 IDLE 模式，0x00:退出 IDLE 模式)
保留位： 0x01 (保留，固定为 0x01)
IDLE 模式空闲时间 IDLE Time：0x03 (3 分钟无操作自动进入 IDLE 模式，范围 0-30 分钟，0x00 表示不自动进入 IDLE 模式)
校验位： 0x0C

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	Enter	保留位	IDLE Time	校验位	帧尾
BB	01	04	00	03	01	01	03	0C	7E

2.31.2 响应帧

如果执行成功，则响应帧为：

帧类型： 0x01
指令代码： 0x04
指令参数长度： 0x0001
指令参数： 0x00（表示执行成功）
校验位： 0x06

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	指令参数	校验位	帧尾
BB	01	04	00	01	00	06	7E

2.32 NXP ReadProtect/Reset ReadProtect 指令

2.32.1 命令帧

NXP G2X 标签支持 ReadProtect/Reset ReadProtect 指令。当标签执行 ReadProtect 指令成功，标签的 ProtectEPC 和 ProtectTID 位将会被设置为’1’，标签会进入到数据保护的状态。如果让标签从数据保护状态回到正常状态，需要执行 Reset ReadProtect 指令。这条指令之前应先设置 Select 参数，以便选择指定的标签进行操作。

帧类型： 0x00
指令代码： 0xE1

指令参数长度： 0x0005

Access Password： 0x0000FFFF

ReadProtect/Reset ReadProtect： 0x00 (0x00 代表执行 ReadProtect，0x01 代表执行 Reset ReadProtect)

校验位： 0x0B

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	AP（MSB）			AP（LSB）	Reset	校验位	帧尾
BB	00	E1	00	05	00	00	FF	FF	00	E4	7E

2.32.2 响应帧

如果 ReadProtect 指令执行正确，则响应帧为：

帧类型： 0x01

指令代码： 0xE1

指令参数长度： 0x0010

操作的标签 PC+EPC 长度 UL：0x0E

操作标签 PC： 0x3000

操作标签 EPC： 0x30751FEB705C5904E3D50D70

指令参数： 0x00（执行成功）

校验位： 0x3D

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	UL	PC（MSB）	PC（LSB）
BB	01	E1	00	10	0E	30	00

EPC（MSB）									
30	75	1F	EB	70	5C	59	04	E3	D5	0D

EPC（LSB）	指令参数	校验位	帧尾
70	00	3D	7E

如果 Reset ReadProtect 指令执行正确，则响应帧为：

帧类型： 0x01

指令代码： 0xE2

指令参数长度： 0x0010

操作的标签 PC+EPC 长度 UL：0x0E

操作标签 PC： 0x3000

操作标签 EPC： 0x30751FEB705C5904E3D50D70

指令参数： 0x00（执行成功）

校验位： 0x3E

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	UL	PC（MSB）	PC（LSB）
BB	01	E2	00	10	0E	30	00

EPC（MSB）									
30	75	1F	EB	70	5C	59	04	E3	D5	0D

EPC（LSB）	指令参数	校验位	帧尾
70	00	3E	7E

如果在执行 ReadProtect(Set/Reset 参数为 0x00)指令的时候，该标签没有在场区，指定的 EPC 代码不对或者标签没有响应，会返回错误代码 0x2A，如下：

帧类型： 0x01
指令代码： 0xFF
指令参数长度： 0x0001
指令参数： 0x2A
校验位： 0x2B

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	指令参数	校验位	帧尾
BB	01	FF	00	01	2A	2B	7E

如果在执行 Reset ReadProtect(Set/Reset 参数为 0x01)指令的时候，该标签没有在场区或者指定的 EPC 代码不对，会返回错误代码 0x2B，如下：

帧类型： 0x01
指令代码： 0xFF
指令参数长度： 0x0001
指令参数： 0x2B
校验位： 0x2C

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	指令参数	校验位	帧尾
BB	01	FF	00	01	2B	2C	7E

如果 Access Password 不正确，则返回错误代码 0x16，并会返回所操作的标签的 PC+EPC，如下：

帧类型： 0x01
指令代码： 0xFF
指令参数长度： 0x0016
指令参数： 0x16
PC+EPC 长度 UL：0x0E
PC： 0x3400
EPC： 0x30751FEB705C5904E3D50D70
校验位： 0x75

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	指令参数	UL	PC（MSB）	PC（LSB）
BB	01	FF	00	10	16	0E	34	00

EPC（MSB）										
30	75	1F	EB	70	5C	59	04	E3	D5	0D

EPC（LSB）	校验位	帧尾
70	75	7E

2.33 NXP Change EAS 指令

2.33.1 命令帧

NXP G2X 标签支持 Change EAS 指令。当标签执行 Change EAS 指令成功，标签的 PSF 位将会相应的变成’1’或者’0’。当标签的 PSF 位置为’1’的时候，标签将响应 EAS_Alarm 指令，否则标签不响应 EAS_Alarm 指令。这条指令之前应先设置 Select 参数，以便选择指定的标签进行操作。

Change EAS 指令帧定义如下：

帧类型： 0x00
指令代码： 0xE3
指令参数长度： 0x0005
Access Password： 0x0000FFFF
Set/Reset： 0x0002
校验位： 0x45

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	AP（MSB）			AP（LSB）	PSF	校验位	帧尾
BB	00	E3	00	05	00	00	FF	FF	01	E7	7E

2.33.2 响应帧

如果 Change EAS 指令执行正确，则响应帧为：

帧类型： 0x01
指令代码： 0xE3
指令参数长度： 0x0010
操作的标签 PC+EPC 长度 UL：0x0E
操作标签 PC： 0x3000
操作标签 EPC： 0x30751FEB705C5904E3D50D70
返回数据： 0x00（执行成功）
校验位： 0x3F

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	UL	PC（MSB）	PC（LSB）
BB	01	E3	00	10	0E	30	00

EPC（MSB）										
30	75	1F	EB	70	5C	59	04	E3	D5	0D

EPC（LSB）	指令参数	校验位	帧尾
70	00	3F	7E

如果在执行 Change EAS 指令的时候，该标签没有在场区，指定的 EPC 代码不对或者标签没有响应，会返回错误代码 0x1B，如下：

帧类型： 0x01
指令代码： 0xFF
指令参数长度： 0x0001
指令参数： 0x1B
校验位： 0x1C

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	指令参数	校验位	帧尾
BB	01	FF	00	01	1B	1C	7E

如果 Access Password 不正确，则返回错误代码 0x16，并会返回所操作的标签的 PC+EPC，如下：

帧类型： 0x01
指令代码： 0xFF
指令参数长度： 0x0016
指令参数： 0x16
PC+EPC 长度 UL：0x0E
PC： 0x3400
EPC： 0x30751FEB705C5904E3D50D70
校验位： 0x75

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	指令参数	UL	PC（MSB）	PC（LSB）
BB	01	FF	00	10	16	0E	34	00
EPC（MSB）								
30	75	1F	EB	70	5C	59	04	E3	D5	0D
EPC（LSB）				校验位			帧尾	
70				75			7E	

2.34 NXP EAS_Alarm 指令

2.34.1 命令帧

NXP G2X 标签支持 EAS_Alarm 指令。当标签接收到 EAS_Alarm 指令后，标签会立刻返回 64bits EAS-Alarm code。注意只有当标签的 PSF 位置为‘1’的时候，标签才响应 EAS_Alarm 指令，否则标签不响应 EAS_Alarm 指令。该指令适合于电子商品防窃（盗）系统。

帧类型： 0x00
指令代码： 0xE4
指令参数长度： 0x0000
校验位： 0xE4

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	校验位	帧尾
BB	00	E4	00	00	E4	7E

2.34.2 响应帧

如果 EAS_Alarm 指令执行成功，有标签响应并返回了正确的 64bits EAS-Alarm code，则响应帧为：

帧类型： 0x01
指令代码： 0xE4
指令参数长度： 0x0001

指令参数： 0x00
校验位： 0x80

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）
BB	01	E4	00	08

EAS-Alarm code（MSB）							EAS-Alarm code（LSB）	校验位	帧尾
69	0A	EC	7C	D2	15	D8	F9	80	7E

如果在执行 EAS_Alarm 指令的时候，没有标签响应，会返回错误代码 0x1D，如下：
帧类型： 0x01
指令代码： 0xFF
指令参数长度： 0x0001
指令参数： 0x1D
校验位： 0x1E

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	指令参数	校验位	帧尾
BB	01	FF	00	01	1D	1E	7E

2.35 NXP ChangeConfig 指令

2.35.1 命令帧

NXP G2X 标签某些系列（如 G2iM 和 G2iM+）支持 ChangeConfig 指令，可以通过该指令读取或修改 NXP G2X 标签的 16bits Config-Word。NXP G2X 标签的 Config-Word 位于标签存储区 Bank 01（即 EPC 区）地址 20h 处（word address），可以通过普通的 Read 指令读取。当标签处于 Secured 状态（安全状态）时，可以改写标签的 Config-Word，需要注意的是改写 Config-Word 是翻转 Config-Word 的相应数据位，即写入‘1’的对应位翻转（‘1’变成‘0’，‘0’变成‘1’），写入‘0’的对应位保持不变。这条指令之前应先设置 Select 参数，以便选择指定的标签进行操作。

帧类型： 0x00
指令代码： 0xE0
指令参数长度： 0x0006
Access Password：0x0000FFFF
Config-Word： 0x0000 (全 0 时标签返回未更改的 Config-Word，相当于读取)
校验位： 0xE4

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	AP（MSB）		
BB	00	E0	00	06	00	00	FF

AP（LSB）	Config（MSB）	Config（LSB）	校验位	帧尾
FF	00	00	E4	7E

2.35.2 响应帧

如果 ChangeConfig 指令执行正确，则响应帧为：

帧类型： 0x01
指令代码： 0xE0

指令参数长度： 0x0011
操作的标签 PC+EPC 长度 UL：0x0E
操作标签 PC： 0x3000
操作标签 EPC： 0x30751FEB705C5904E3D50D70
Config-Word： 0x0041
校验位： 0x7E

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	UL	PC（MSB）	PC（LSB）
BB	01	E0	00	11	0E	30	00

EPC（MSB）										
30	75	1F	EB	70	5C	59	04	E3	D5	0D

EPC（LSB）	Config（MSB）	Config（LSB）	校验位	帧尾
70	00	41	7E	7E

如果在执行 ChangeConfig 指令的时候，该标签没有在场区，指定的 EPC 代码不对或者标签没有响应，会返回错误代码 0x1A，如下：
帧类型： 0x01
指令代码： 0xFF
指令参数长度： 0x0001
指令参数： 0x1A
校验位： 0x1B

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	指令参数	校验位	帧尾
BB	01	FF	00	01	1A	1B	7E

如果 Access Password 不正确，则返回错误代码 0x16，并会返回所操作的标签的 PC+EPC，如下：
帧类型： 0x01
指令代码： 0xFF
指令参数长度： 0x0016
指令参数： 0x16
PC+EPC 长度 UL：0x0E
PC： 0x3400
EPC： 0x30751FEB705C5904E3D50D70
校验位： 0x75

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	指令参数	UL	PC（MSB）	PC（LSB）
BB	01	FF	00	10	16	0E	34	00

EPC（MSB）										
30	75	1F	EB	70	5C	59	04	E3	D5	0D

EPC（LSB）	校验位	帧尾
70	75	7E

2.36 Impinj Monza QT 指令

2.36.1 命令帧

Impinj Monza 4 QT 标签支持 QT 指令，该指令可以修改标签的 QT Control word，其中设置 QT_SR 位可以缩短标签在 Open（开放）和 Secured（安全）状态或者即将进入到 Open 和 Secured 状态时的操作距离，修改 QT_MEM 位可以切换标签使用 Public Memory Map（公共存储区）还是 Private Memory Map（私有存储区）。这条指令之前应先设置 Select 参数，以便选择指定的标签进行操作。

QT 指令帧定义如下，在本例中是设置 QT_MEM 位为 1 并写入标签非挥发性存储区：

帧类型： 0x00
指令代码： 0xE5
指令参数长度： 0x0008
Access Password：0x0000FFFF
Read/Write： 0x01 (0x00：Read，0x01：Write)
Persistence： 0x01 (0x00：写入标签挥发性存储区，0x01：写入非挥发性存储区)
Payload： 0x4000(QT Control，最高两个 bits 分别为 QT_SR 和 QT_MEM)
校验位： 0x2D

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	AP（MSB）		
BB	00	E5	00	08	00	00	FF

AP（LSB）	Read/Write	Persistence	Payload0	Payload1	校验位	帧尾
FF	01	01	40	00	2D	7E

2.36.2 响应帧

如果 QT 指令执行正确，当 Read/Write 数据域为 0x00 时，响应帧为：

帧类型： 0x01
指令代码： 0xE5
指令参数长度： 0x0011
操作的标签 PC+EPC 长度 UL：0x0E
操作标签 PC： 0x3000
操作标签 EPC： 0x30751FEB705C5904E3D50D70
QT Control Word： 0x0000
校验位： 0x42

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	UL	PC（MSB）	PC（LSB）
BB	01	E5	00	11	0E	30	00

EPC（MSB）										
30	75	1F	EB	70	5C	59	04	E3	D5	0D

EPC（LSB）	QT Control0	QT Control1	校验位	帧尾
70	00	00	42	7E

如果 QT 指令执行正确，当 Read/Write 数据域为 0x01 时，响应帧为：

帧类型： 0x01
指令代码： 0xE6

指令参数长度： 0x0010
操作的标签 PC+EPC 长度 UL：0x0E
操作标签 PC： 0x3000
操作标签 EPC： 0x30751FEB705C5904E3D50D70
指令参数： 0x00
校验位： 0x42

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	UL	PC（MSB）	PC（LSB）
BB	01	E6	00	10	0E	30	00

EPC（MSB）										
30	75	1F	EB	70	5C	59	04	E3	D5	0D

EPC（LSB）	指令参数	校验位	帧尾
70	00	42	7E

如果在执行 QT 指令的时候，该标签没有在场区，指定的 EPC 代码不对或者标签没有响应，会返回错误代码 0x2E，如下：
帧类型： 0x01
指令代码： 0xFF
指令参数长度： 0x0001
指令参数： 0x2E
校验位： 0x2F

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	指令参数	校验位	帧尾
BB	01	FF	00	01	2E	2F	7E

如果 Access Password 不正确，则返回错误代码 0x16，并会返回所操作的标签的 PC+EPC，如下：
帧类型： 0x01
指令代码： 0xFF
指令参数长度： 0x0016
指令参数： 0x16
PC+EPC 长度 UL：0x0E
PC： 0x3400
EPC： 0x30751FEB705C5904E3D50D70
校验位： 0x75

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	指令参数	UL	PC（MSB）	PC（LSB）
BB	01	FF	00	10	16	0E	34	00

EPC（MSB）										
30	75	1F	EB	70	5C	59	04	E3	D5	0D

EPC（LSB）	校验位	帧尾
70	75	7E

2.37 BlockPermalock 指令

2.37.1 命令帧

BlockPermalock 指令可以永久锁定用户区的某几个 Block，或者读取 Block 的锁定状态。这条指令之前应先设置 Select 参数，以便选择指定的标签进行操作。
BlockPermalock 指令帧定义如下，在本例中是 BlockPermalock 状态写入，将第 5、6、7 个 Block 进行永久锁定：

帧类型： 0x00
指令代码： 0xD3
指令参数长度： 0x0009
Access Password：0x0000FFFF
Read/Lock： 0x00 (0x00：Read，0x01：Lock)
BlockPtr： 0x0000 (Mask 的起始 Block 地址，以 16 个 Block 为单位)
BlockRange： 0x01 (16 个 Block 为单位)
Mask： 0x0700 (当 Read/Lock 数据域为 0x00，即读取状态时，该数据域省略)
校验位： 0xE8

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	AP（MSB）			AP（LSB）
BB	00	D3	00	0B	00	00	FF	FF

Read/Lock	MemBank	BlockPtr1	BlockPtr0	BlockRange	Mask（MSB）	Mask（LSB）	校验位	帧尾
01	03	00	00	01	07	00	E8	7E

2.37.2 响应帧

如果 BlockPermalock 指令执行正确，当 Read/Lock 数据域为 0x00 时，响应帧为：

帧类型： 0x01
指令代码： 0xD3
指令参数长度： 0x0012
操作的标签 PC+EPC 长度 UL：0x0E
操作标签 PC： 0x3000
操作标签 EPC： 0xE20030166606006911609F94
BlockRange： 0x01
BlockPermalock 状态： 0x0700
校验位： 0xCD

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	UL	PC（MSB）	PC（LSB）
BB	01	D3	00	12	0E	30	00

EPC（MSB）										
E2	00	30	16	66	06	00	69	11	60	9F

EPC（LSB）	BlockRange	返回数据（MSB）	返回数据（LSB）	校验位	帧尾
70	01	07	00	CD	7E

如果 BlockPermalock 指令执行正确，当 Read/Lock 数据域为 0x01 时，响应帧为：

帧类型： 0x01
指令代码： 0xD4
指令参数长度： 0x0010
操作的标签 PC+EPC 长度 UL：0x0E
操作标签 PC： 0x3000
操作标签 EPC： 0xE20030166606006911609F94
指令参数： 0x00（执行成功）
校验位： 0xC4

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	UL	PC（MSB）	PC（LSB）
BB	01	D4	00	10	0E	30	00

EPC（MSB）										
E2	00	30	16	66	06	00	69	11	60	9F

EPC（LSB）	指令参数	校验位	帧尾
94	00	C4	7E

如果在执行 BlockPermalock 指令的时候，该标签没有在场区，指定的 EPC 代码不对或者标签没有响应，会返回错误代码 0x14，如下：

帧类型： 0x01
指令代码： 0xFF
指令参数长度： 0x0001
指令参数： 0x14
校验位： 0x15

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	指令参数	校验位	帧尾
BB	01	FF	00	01	14	15	7E

如果在执行 BlockPermalock 指令的时候，操作标签返回了 EPC Gen2 协议规定的错误代码(error-code)，因为 EPC Gen2 规定的错误代码只有低 4 位有效，响应帧会将标签返回的错误代码或上 0xE0 之后再返回。比如如果发送指令参数 BlockPtr 超过标签数据存储区的 Block 范围，标签会返回错误代码 0x03(存储区超出，Memory Overrun)。响应帧则返回错误代码 0xE3，并返回所操作标签的 PC+EPC，如下：

帧类型： 0x01
指令代码： 0xFF
指令参数长度： 0x0010
指令参数： 0xA3
PC+EPC 长度 UL：0x0E
PC： 0x3000
EPC： 0xE20030166606006911609F94
校验位： 0xD2

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	错误代码	UL	PC（MSB）	PC（LSB）
BB	01	FF	00	10	E3	0E	30	00

EPC（MSB）										
E2	00	30	16	66	06	00	69	11	60	9F

EPC（LSB）	校验位	帧尾
94	D2	7E

如果 Access Password 不正确，则返回错误代码 0x16，并会返回所操作的标签的 PC+EPC，如下：

帧类型： 0x01

指令代码： 0xFF

指令参数长度： 0x0016

指令参数： 0x16

PC+EPC 长度 UL：0x0E

PC： 0x3000

EPC： 0xE20030166606006911609F94

校验位： 0x05

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	指令参数	UL	PC（MSB）	PC（LSB）
BB	01	FF	00	10	16	0E	34	00

EPC（MSB）										
E2	00	30	16	66	06	00	69	11	60	9F

EPC（LSB）	校验位	帧尾
94	05	7E

3、指令总结

代码	描述
0x03	获取读写器模块信息
0x22	单词轮询指令
0x27	多次轮询指令
0x28	停止多次轮询指令
0x0C	设置 Select 参数指令
0x0B	获取 Select 参数指令
0x12	设置发送 Select 指令
0x39	读标签数据存储区
0x49	写标签数据存储区
0x82	锁定 Lock 标签数据存储区
0x65	灭活 Kill 标签
0x0D	获取 Query 参数
0x0E	设置 Query 参数
0x07	设置工作地区
0xAB	设置工作信道
0xAA	获取工作信道
0xAD	设置自动跳频
0xB7	获取发射功率
0xB6	设置发射功率
0xB0	设置发射连续载波
0xF1	获取接收解调器参数
0xF0	设置接收解调器参数
0xF2	测试射频输入端阻塞信号
0xF3	测试信道 RSSI
0x1A	控制 IO 端口
0x17	模块休眠
0x1D	设置模块空闲休眠时间
0xE0	NXP ChangeConfig 指令
0xE1	NXP ReadProtec/Reset ReadProtect 指令
0xE3	NXP Change EAS 指令
0xE4	NXP EAS-Alarm 指令
0xE5/0xE6	Impinj Monza 4 QT 指令
0xD3/0xD4	BlockPermalock 指令

4、命令帧执行失败的响应帧总结

如果命令帧执行失败，则 M100 芯片向上位机发送执行失败的响应帧。执行失败的响应帧共用指令代码 0xFF。如果在执行失败之前没有得到标签的 EPC，则指令参数固定为 1 个 byte 的错误代码。如果在执行失败前得到了标签的 EPC，则响应帧参数为 1 个 byte 的错误代码再加上标签的 PC+EPC 数据。

例如，如果轮询命令帧执行失败，没有收到标签返回或者返回数据 CRC 校验错误，将返回错误代码 0x15，如下：

帧类型： 0x01

指令代码： 0xFF (0xFF 代表命令帧执行失败)

指令参数长度： 0x01

指令参数： 0x15 (为执行失败后返回的错误代码)

校验位： 0x16

帧头	帧类型	指令代码	指令参数长度（MSB）	指令参数长度（LSB）	指令参数	校验位	帧尾
BB	01	FF	00	01	15	16	7E

错误代码总结如下:

类型	代码	描述
Command Error	0x17	命令帧中指令代码错误。
FHSS Fail	0x20	跳频搜索信道超时。所有信道在这段时间内都被占用。
Inventory Fail	0x15	轮询操作失败。没有标签返回或者返回数据 CRC 校验错误。
Access Fail	0x16	访问标签失败，有可能是访问密码 password 不对。
Read Fail	0x09	读标签数据存数区失败。标签没有返回或者返回数据 CRC 校验错误。
Read Error	0xA0 | Error code	读标签数据存储区错误。返回的代码由 0xA0 位或 Error Code 得到。Error code 信息详见下表。
Write Fail	0x10	写标签数据存数区失败。标签没有返回或者返回数据 CRC 校验错误。
Write Error	0xB0 | Error code	写标签数据存储区错误。返回的代码由 0xB0 位或 Error Code 得到。Error code 信息详见下表。
Lock Fail	0x13	锁定标签数据存数区失败。标签没有返回或者返回数据 CRC 校验错误。
Lock Error	0xC0 | Error code	锁定标签数据存储区错误。返回的代码由 0xC0 位或 Error Code 得到。Error code 信息详见下表。
Kill Fail	0x12	灭活标签失败，标签没有返回或者返回数据 CRC 校验错误。
Kill Error	0xD0 | Error code	灭活标签错误。返回的代码由 0xC0 位或 Error Code 得到。Error code 信息详见 EPC Gen2 协议中标签返回错误代码。
BlockPermalock Fail	0x14	BlockPermalock 执行失败。标签没有返回或者返回数据 CRC 校验错误。
BlockPermalock Error	0xE0 | Error code	BlockPermalock 错误。返回的代码由 0xE0 位或 Error Code 得到。Error code 信息详见 EPC Gen2 协议中标签返回错误代码。

NXP G2X 标签特有指令错误代码：

类型	代码	描述
ChangeConfig Fail	0x1A	ChangeConfig 指令失败，标签没有返回数据或者返回数据 CRC 校验错误。
ReadProtect Fail	0x2A	ReadProtect 指令失败，标签没有返回数据或者返回数据 CRC 校验错误。
Reset ReadProtect Fail	0x2B	Reset ReadProtect 指令失败，标签没有返回数据或者返回数据 CRC 校验错误。
Change EAS Fail	0x1B	Change EAS 指令失败，标签没有返回数据或者返回数据 CRC 校验错误。
EAS_Alarm Fail	0x1D	EAS_Alarm 指令失败，没有标签返回正确 Alarm Code。
特有指令标签返回的错误代码	0xE0 | Error code	特有指令标签返回的错误代码，错误代码由 0xE0 或上标签返回的 Error Code 得到。

Impinj Monza QT 标签特有指令错误代码：

类型	代码	描述
QT Fail	0x2E	QT 指令失败，标签没有返回数据或者返回数据 CRC 校验错误。
特有指令标签返回的错误代码	0xE0 | Error code	特有指令标签返回的错误代码，错误代码由 0xE0 或上标签返回的 Error Code 得到。

EPC Gen2 协议中标签返回错误代码：

Tag error-code

Error-code Support	Error Code	Error Code Name	Error Description
Error-specific	00000000_2	Other error	本表中没有声明的其他所有错误。
	00000011_2	Memory overrun	指定的标签数据存储区不存在；或者该标签不支持指定长度的 EPC，比如 XPC。
	00000100_2	Memory locked	指定的标签数据存储区被锁定并且/或者是永久锁定，而且锁定状态为不可写或不可读。
	00001011_2	Insufficient power	标签没有收到足够的能量来进行写操作。
Non-specific	00001111_2	Non-specific error	标签不支持 Error-code 返回。