Q240/EN11 系列 RFID 读写头 TCP/IP 通讯协议

Version 1.3 ， 2019/04/26

1. Q240 RFID 读写头自定义应用层数据帧

Addr 2Byte	包头 1Byte	帧长度 1Byte	命令 1Byte	状态 1Byte	数据 Buffer	包尾 1Byte
Reserve	0xA0					0xE0

1. 1. TCP 端口号

9000

1. 2. Addr(2Byte)

预留作为 UART 方式使用，源地址

1. 3. 帧长度(1Byte)

包含协议所有字段，从 Addr 到包尾

1. 4. 命令(1Byte)

一共包含 11 种命令：

命令	功能
0xA1	功率设置
0xA2	功率读取
0xA3	读标签
0xA4	写标签
0xA5	区域设置
0xA6	区域读取
0xA7	频率设置
0xA8	频率读取
0xA9	单次寻卡
0xAA	连续寻卡
0xAB	停止连续寻卡
0XB1	连续扫卡输出过滤模式设置

1. 5. 状态(1Byte)

用于标记返回状态，发送命令该字段默认 0x00。
返回状态描述：
0xC2：操作完成
0xC8：操作失败，发生错误

1. 6. 数据 Buffer

长度不固定，由具体命令而定，根据不同命令，存放相关数据，具体描述如下：
发送帧：设置参数数据，需要写入的数据。
返回帧：读取的设置参数数据，读取的用户数据，错误代码。

错误代码描述如下：

错误代码	描述
0x01	与 RFID MCU 通讯失败
0x02	读取失败
0x03	写入失败
0x04	设置参数失败
0x05	工作状态异常，正在进行连续扫卡，请先停止连续寻卡操作
0x06	命令错误
0x10	异常错误

2. 命令操作方法

RFID 读写头操作模式采用一问一答式，由主机（上位机）发起，RFID 读写头不会主动发起只是应答，由主机（上位机）进行流量控制。

2. 1. 功率设置

上位机至读写头：发送功率设置命令，直至收到操作完成或错误状态。

保留		包头	帧长度	命令	状态	保留	读功率	写功率	包尾
0x00	0x00	0xA0	0x0A	0xA1	0x00	0x00			0xE0

读写头至上位机：收到命令并应答。

保留		包头	帧长度	命令	状态	错误代码 *	包尾
0x00	0x00	0xA0	0x08	0xA1			0xE0

■ 注意：

✧ 功率设置范围：5 dBm ~30dBm，使用十六进制表示，即 0x05~0x1E。
✧ 返回帧中，状态为 C2 时，错误代码为 0x00，状态为 C8 时，显示相应错误代码。

示例：

设置读功率 15dBm，写功率 30dBm，发送：
00 00 A0 0A A1 00 00 0F 1E E0
返回：
00 00 A0 08 A1 C2/C8 00 E0

2. 2. 读取功率设置值

上位机至读写头：发送读取功率命令，直至收到操作完成或错误状态。

保留		包头	帧长度	命令	状态	包尾
0x00	0x00	0xA0	0x07	0xA2	0x00	0xE0

读写头至上位机：收到命令并应答，返回功率值数据。

保留		包头	帧长度	命令	命令	保留	读功率	写功率	包尾
0x00	0x00	0xA0	0x0A	0xA2	0x00	0x00			0xE0

示例：

发送读取功率命令：
00 00 A0 07 A2 00 E0
返回：
00 00 A0 0A A2 C2 00 0F 1E E0
读取到的功率：读功率 15dBm，写功率 30dBm。

2. 3. 区域设置

上位机至读写头：发送区域设置命令，直至收到操作完成或错误状态。

保留		包头	帧长度	命令	状态	区域代码	包尾
0x00	0x00	0xA0	0x08	0xA5	0x00		0xE0

读写头至上位机：收到命令并应答。

保留		包头	帧长度	命令	状态	错误代码 *	包尾
0x00	0x00	0xA0	0x08	0xA5			0xE0

■ 注意：

✧ 区域代码描述：
0x00——USA　0x01——China1　0x02——China2
0x03——Europe　0x04——Korea　0x05——Japan
✧ 区域设置后，发射频率段为相应区域的频段。
✧ 返回帧中，状态为 C2 时，错误代码为 0x00，状态为 C8 时，显示相应错误代码。

示例：

设置欧洲地区，发送：
00 00 A0 08 A5 00 03 E0

返回：
00 00 A0 08 A5 C2/C8 00 E0

2.4. 区域代码读取

上位机至读写头：发送读取参数命令，直至收到设置完成或错误状态。

保留		包头	帧长度	命令	状态	包尾
0x00	0x00	0xA0	0x07	0xA6	0x00	0xE0

读写头至上位机：收到命令并应答，返回参数数据。

保留		包头	帧长度	命令	状态	区域代码	包尾
0x00	0x00	0xA0	0x08	0xA6	0x00		0xE0

示例：

发送读取区域代码命令：
00 00 A0 07 A6 00 E0
返回：
00 00 A0 08 A6 C2 03 E0
区域代码是欧洲。

2.5. 频率设置

上位机至读写头：发送设置参数命令，直至收到操作完成或错误状态。

保留		包头	帧长度	命令	状态	频点数	频点 1			…	包尾
0x00	0x00	0xA0	Len	0xA7	0x00	NUM					0xE0

读写头至上位机：收到命令并应答。

保留		包头	帧长度	命令	状态	错误代码 *	包尾
0x00	0x00	0xA0	0x08	0xA7			0xE0

■ 注意：

- ✧ 频率设置包括跳频频点数和跳频频点列表，一次最多设置 32 个跳频频点，每个跳频点单位为 KHz，转换为 16 进制数，使用 3 字节表示，920.125MHz=920125KHz=0E 0A 3D。
- ✧ Len=NUM*3+8。
- ✧ 返回帧中，状态为 C2 时，错误代码为 0x00，状态为 C8 时，显示相应错误代码。

示例：

设置 5 个跳频频点，分别为：920.125(0E0A3D)、921.250(0E0EA2)、921.625(0E1019)、

922.375(0E1307)、924.375(0E1AD7)

00 00 A0 17 A7 00 05 0E 0A 3D 0E 0E A2 0E 10 19 0E 13 07 0E 1A D7 E0

返回：

00 00 A0 08 A7-C2/C8 00 E0

2.6. 频率读取

上位机至读写头：发送读取参数命令，直至收到操作完成或错误状态。

保留		包头	帧长度	命令	状态	包尾
0x00	0x00	0xA0	0x07	0xA8	0x00	0xE0

读写头至上位机：收到命令并应答，返回参数数据。

保留		包头	帧长度	命令	状态	频点数	频点 1			…	包尾
0x00	0x00	0xA0	Len	0xA8	0x00	NUM					0xE0

示例：

发送读取频率命令：

00 00 A0 07 A8 00 E0

返回：

00 00 A0 17 A8 C2 05 0E 0A 3D 0E 0E A2 0E 10 19 0E 13 07 0E 1A D7 E0

读取到的频率：

5 个跳频频点分别为：920.125(0E0A3D)、921.250(0E0EA2)、921.625(0E1019)、922.375(0E1307)、924.375(0E1AD7)。

2.7. 读标签操作

上位机至读写头：发送读标签命令，直至收到操作完成或错误状态。

保留		包头	帧长度	命令	状态	Bank	Addr	Data Len	*Filter Bank	*Filter Addr	*Filter Data Len	*Filter Data	包尾
0x00	0x00	0xA0	0x0A	0xA3	0x00								0xE0

读写头至上位机：收到命令并应答。

保留		包头	帧长度	命令	状态	Bank	Addr	DataLen	Data	包尾
0x00	0x00	0xA0	Len	0xA3					……	0xE0

■ 注意：

- ✧ 标签 Bank 分区：0x01—EPC，0x02—TID，0x03—USER；
- ✧ 由于数据以字为单位存储，所以 Addr 和 DataLen 只能为偶数；
- ✧ 一次最多读取 32 个字节数据；

- ✧ Len=Data Len+10。
- ✧ 数据过滤功能，指可按照指定要求读取特定标签的数据，根据设定的过滤条件，只读取满足相应条件的标签的内容，其它标签不响应。可选，如果使用，协议中 filter 相关字段生效，如果不使用，协议中缺省相应字段。FilterBank：特定 bank，0x01—EPC,0x02—TID,0x03—USR；FilterAddr：特定起始位置（以字节为单位）；FilterDataLen：特定数据长度（以字节为单位）；FilterData：特定数据。

示例：

1).无过滤功能，读取 USR 区，起始地址 0x00，读取长度 4 个字节

00 00 A0 0A A3 00 03 00 04 E0
返回
00 00 A0 0E A3 C2 03 00 04 01 02 03 04 E0

2).使用过滤功能，指定 EPC=0x E2009A3060034AF000001251 的标签（EPC 有效值为 EPC bank 区 0x04 地址起，长度 12），读取 USR 区，起始地址 0x00，读取长度 4 个字节

00 00 A0 19 A3 00 03 00 04 01 04 0C E2 00 9A 30 60 03 4A F0 00 00 12 51 E0
返回
00 00 A0 0E A3 C2 03 00 04 01 02 03 04 E0

2. 8. 写标签操作

上位机至读写头：发送写操作命令并随带标签数据，直至收到操作完成或错误状态。

保留		包头	帧长度	命令	状态	Bank	start Addr	Data Len	Data
0x00	0x00	0xA0	Len	0xA4					……

*Filter Bank	*Filter Addr	*Filter DataLen	*Filter Data	包尾
				0xE0

读写头至上位机：收到写操作命令并应答。

保留		包头	帧长度	命令	状态	错误代码 *	包尾
0x00	0x00	0xA0	0x08	0xA4			0xE0

■ 注意：

- ✧ 标签 Bank 分区：0x01—EPC，0x02—TID，0x03—USER；一般只有 USR 区可写；
- ✧ 由于数据以字为单位存储，所以 Addr 和 DataLen 只能为偶数
- ✧ 一次最多写入 32 个字节数据
- ✧ 无过滤功能：Len=DataLen+10 使用过滤：Len=DataLen+10+3+ Filter Data Len

✧ 返回帧中，状态为 C2 时，错误代码为 0x00，状态为 C8 时，显示相应错误代码

✧ 数据过滤功能，指可按照指定要求写入特定标签的数据，根据设定的过滤条件，只写入满足相应条件的标签的内容，其它标签不响应。可选，如果使用，协议中 filter 相关字段生效，如果不使用，协议中缺省相应字段。FilterBank：特定 bank，0x01—EPC,0x02—TID,0x03—USR；FilterAddr：特定起始位置（以字节为单位）；FilterDataLen：特定数据长度（以字节为单位）；FilterData：特定数据。

✧

示例：

1).不过滤，写入 USR 区，起始地址 0x00，写入长度 4 个字节，0x01 0x02 0x03 0x04:
00 00 A0 0E A4 00 03 00 04 01 02 03 04 E0
返回
00 00 A0 08 A4 C2/C8 00 E0

2).使用过滤功能，指定 EPC=0x E2009A3060034AF000001251 的标签（EPC 有效值为 EPC bank 区 0x04 地址起，长度 12），写入 USR 区，起始地址 0x00，写入长度 4 个字节，0x01 0x02 0x03 0x04

00 00 A0 1D A4 00 03 00 04 01 02 03 04 01 04 0C E2 00 9A 30 60 03 4A F0 00 00 12 51 E0
返回
00 00 A0 08 A4 C2/C8 00 E0

2. 9. 单次寻卡

上位机至读写头：发送单次寻卡命令，直至收到操作完成或错误状态。

保留		包头	帧长度	命令	状态	包尾
0x00	0x00	0xA0	0x07	0xA9	0x00	0xE0

读写头至上位机：收到寻卡命令并应答。

保留		包头	帧长度	命令	状态	保留	标签数据	包尾
0x00	0x00	0xA0	0x18	0xA9		0x00	16Byte	0xE0

■ **注意：**

✧ 标签数据=PC+EPC+RSSI，RSSI 以补码形式表示，共 16bit，为实际值*10，如-65.7dBm，则 RSSI=FD6F

示例：

发送单次寻卡指令：
00 00 A0 07 A9 00 E0
返回：
00 00 A0 18 A8 C2 00 34 00 E2 00 9A 30 60 03 4A F0 00 00 12 51 FC 93 E0

读取到的标签数据：
PC=0x3400，EPC=0x E2009A3060034AF000001251，RSSI=-87.7dBm

2.10. 连续寻卡

上位机至读写头：发送连续寻卡命令

保留		包头	帧长度	命令	状态	包尾
0x00	0x00	0xA0	0x07	0xAA	0x00	0xE0

读写头至上位机：收到连续寻卡命令并应答

保留		包头	帧长度	命令	状态	保留	标签数据	包尾
0x00	0x00	0xA0	0x18	0xAA		0x00	16Byte	0xE0

■ 注意：

✧ 返回内容与单次寻卡一致，但会连续返回扫到的所有标签信息
✧ 连续扫卡过程中，只能响应停止连续扫卡命令，不响应其它命令。在执行其它命令之前，请先停止连续扫卡。

示例：

发送连续寻卡指令：
00 00 A0 07 AA 00 E0
返回：
00 00 A0 18 A8 C2 00 34 00 E2 00 9A 30 60 03 4A F0 00 00 12 51 FC 93 E0
00 00 A0 18 A8 C2 00 34 00 E2 00 9A 30 60 03 4A F0 00 00 12 52 FC 93 E0
00 00 A0 18 A8 C2 00 34 00 E2 00 9A 30 60 03 4A F0 00 00 12 54 FC 93 E0
扫到 3 个不同标签。

2.11. 停止连续寻卡

上位机至读写头：发送停止连续寻卡命令

保留		包头	帧长度	命令	状态	包尾
0x00	0x00	0xA0	0x07	0xAB	0x00	0xE0

读写头至上位机：收到停止连续寻卡命令并应答

保留		包头	帧长度	命令	状态	错误代码 *	包尾
0x00	0x00	0xA0	0x08	0xAB			0xE0

示例：

发送停止连续寻卡指令：
00 00 A0 07 AB 00 E0
返回：
00 00 A0 08 AB C2/C8 00 E0

2.12. 连续寻卡输出过滤模式设置

上位机至读写头：发送设置命令

保留		包头	帧长度	命令	状态	模式	包尾
0x00	0x00	0xA0	0x07	0XB1	0x00		0xE0

读写头至上位机：收到停止连续寻卡命令并应答

保留		包头	帧长度	命令	状态	错误代码 *	包尾
0x00	0x00	0xA0	0x08	0XB1			0xE0

■ 注意：

模式：0x00—关闭标签过滤，同一个标签只上传一次（默认状态）
0x01—开启标签过滤，同一个标签扫到几次上传几次。

示例：

开启标签过滤：
00 00 A0 08 B1 00 01 E0
返回
00 00 A0 08 B1 C2/C8 00 E0