

# **WRM200 通信协议（RF433）**

## **—MePanel/MeReceiver/MeWallSwitch**

成都微智慧科技有限公司

（版权所有，翻版必究）

## 目录

一、 协议说明.....	4
二、 报文格式.....	4
1. 报文头格式 .....	4
2. 数据区 .....	5
3. 可选数据区 .....	5
三、 交互流程.....	7
四、 协议内容.....	8
1. 无线无源开关 .....	8
1.1 数据帧.....	8
1.2 无线设备数据域定义 .....	8
1.3 自电设备类型.....	9
2. 有源设备 .....	9
2.1 基本状态查询.....	9
2.2 基本开关控制.....	10
2.3 主动状态上报.....	12
2.4 管理命令.....	15
五、 报文示例.....	17
1. 下发状态查询报文.....	17
2. 下发状态控制报文.....	17
3. 按钮事件上报 .....	17
六、 附录 A.....	18
1. 设备类型代码 .....	18
2. 命令类型代码 .....	18
3. 事件类型代码（仅支持有源设备） .....	18
4. 通道（回路）与开关.....	18
七、 附录 B.....	20
1. CRC8 效验.....	20

## 修订记录

文档版本	修改内容	修改人	日期	备注
1.0	首版发行	Nisary	2017-03-01	
1.1	1.更新交互流程 2.加入管理命令	Nisary	2017-03-02	

## 一、 协议说明

- 本协议作用范围为，基于 WRM200 通信模块的 MePanel, MeWallSwitch, MeRecevier 无线（无源）设备与网关的应用通信通用规约，只进行通信格式和基础通信报文定义。
- 本协议以字节为最小传输单位。
- 协议使用大端模式即数据高字节在前低字节在后。
- 报文采用 CRC8 分别对帧头和数据进行校验。
- 采用 WRM200 通信模块，模块配置为：
  - 波特率：57600
  - 数据位：8 bit
  - 校验位：None
  - 停止位：1
  - 流控：None

## 二、 报文格式

同步字	报文头	报文头校验	数据区	可选数据	数据区校验
-----	-----	-------	-----	------	-------

其中：

- 同步字：约定为 0x55
- 报文头：协议格式定义，实现数据过滤
- 报文头校验：用 CRC8 进行报文头的校验
- 数据区：报文数据主体，承载报文传输数据
- 可选数据区：预留。第一个字节为 TEL——CTRL，其余预留为 0，其中 TEL\_CTRL 定义为（Bit0）：
  - 0：简单透传
  - 1：带重发机制
- 数据校验：报文数据校验(数据区+可选数据区)

### 1. 报文头格式

报文头（3）		
数据区长度（2）	可选数据区长度（1）	报文类型（1）

其中：

- 数据区长度：数据区的字节数
- 可选数据区的长度：可选数据区的字节数
- 报文类型：本协议使用的报文类型如下：

报文类型（1）		
类型值	类型名	说明

0x00	——	保留
0x01	RADIO	无线报文
0x02	RESPONSE	应答
0x03	REPORT	主动上报
0x04	——	保留
0x05	COMMAND	网关模块管理命令
0x06	COMMAND_ACK	网关模块管理命令应答
0x07-0xFF	——	保留

其中网关模块管理命令目前支持：

网关模块管理命令（1）		
命令值	命令名	说明
0x00	——	保留
0x01	PING	PING 报文
0x02	READ_FW_VER	读取射频模块固件版本
0x03-0xFF	——	保留

## 2. 数据区

本协议的数据区定义如下：

数据区（7/9）									
报文状态（1）								地址（4）	类型（1）
B7	B6	B5	B4	B3	B2	B1	B0	源/目地址（取决于类型）	
0	RC								

其中：

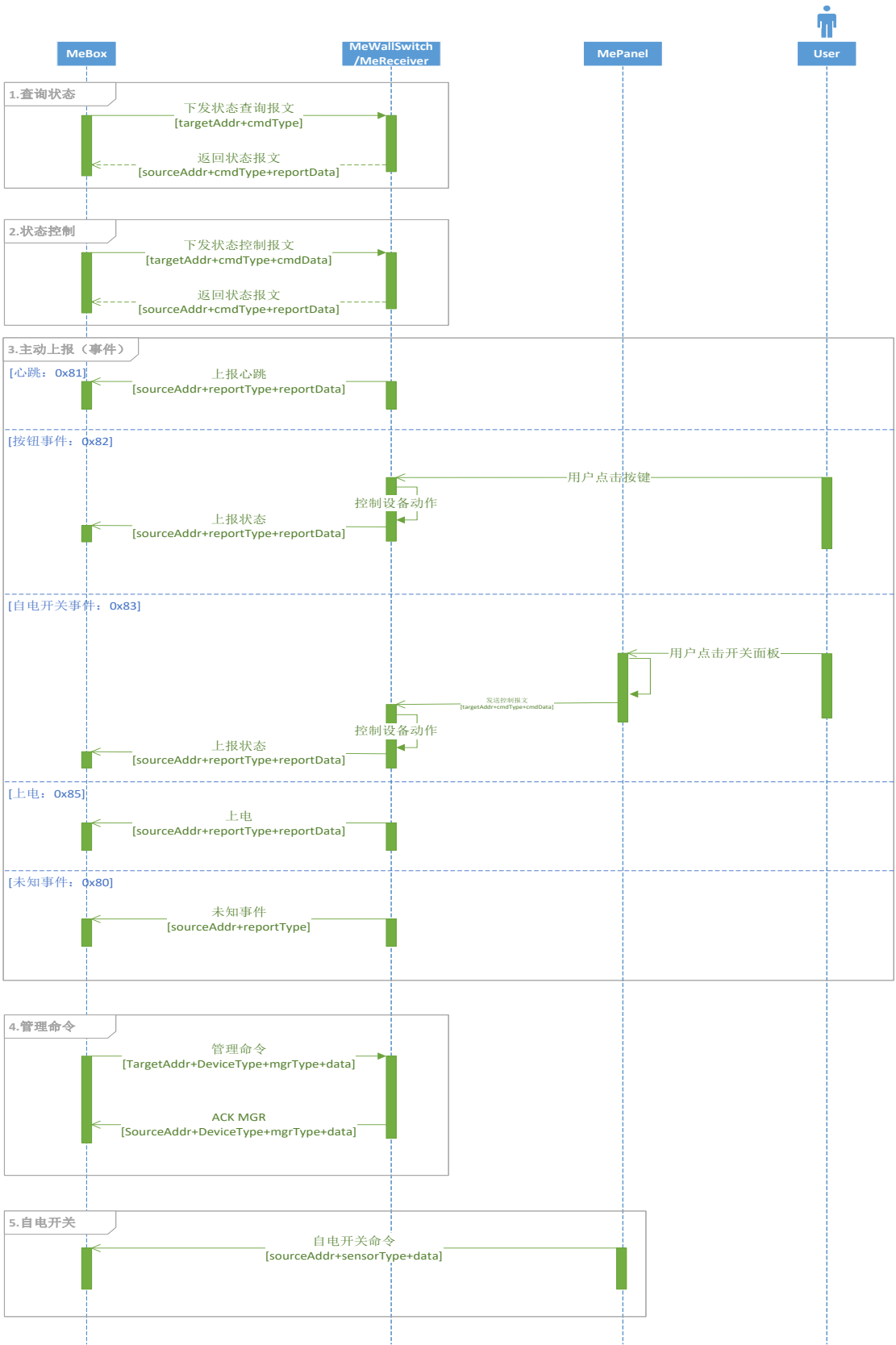
- 上报报文的地址使用 TargetAddr,其他使用 SourceAddr
- 对于无线无源开关设备数据字段仅有 cmd data，其他有源设备才有 cmd type
- RC 表示中继转发：（最多转发两次）
  - 10：没有转发
  - 20：转发了一次
  - 40：转发了两次

## 3. 可选数据区

可选数据区长度（7）	
报文控制类型（1）	保留（6）

上传	下发								0x000000000000
RSSI	TEL_CTRL								
场强值	B7	B6	B5	B4	B3	B2	B1	B0	
	0	0	0	0	0	0	0	0: 简单透传	
								1: 带重发机制	

### 三、 交互流程



## 四、 协议内容

### 1. 无线无源开关

#### 1.1 数据帧

同步字(1)	报文头(3)			报头效验(1)
	数据区长度(2)	可选数据区长度(1)	报文类型(1)	CRC8
0x55	0x0007	0x07	0x01	0xnn

数据区(9)											
报文状态(1)								地址(4)	类型(1)	数据(1)	
B7	B6	B5	B4	B3	B2	B1	B0	目地址	Sensor Type	Data(1)	
0	RC	0	0	0	0	0	0				
0	0	0b00000						SourceAddr	<a href="#">见设备类型代码</a>	<a href="#">见自电开关数据域定义</a>	

可选数据区长度（7）								
报文控制类型（1）								保留（6）
TEL_CTRL								
B7	B6	B5	B4	B3	B2	B1	B0	
0	0	0	0	0	0	0	0：简单透传	0x000000000000
							1：带重发机制	

数据效验（1）	
数据区	可选数据区
CRC8	

#### 1.2 无线设备数据域定义

无线设备数据域(1)							
B7	B6	B5	B4	B3	B2	B1	B0
RESERVE		P/R	I/O	Rocker D-A			
0b00		按下/弹起检测: P=1, R=0	开/关动作指示: 1: I 侧有动作 0: O 侧有动作	每一位分别对应 Rocker 的状态, 有动作为 1, 无动作为 0			



### 1.3 自电设备类型

自电设备类型 (1)	
类型值	类型名
0x11	单开自电开关
0x12	双开自电开关
0x14	四开自电开关
0x17	圆形三键开关

## 2. 有源设备

### 2.1 基本状态查询

用户获取设备的基本开关状态（多路负载可获取每一路的开关状态）

#### 2.1.1 主机发送的数据格式

同步字(1)	报文头 (3)			报头校验 (1)
	数据区长度 (2)	可选数据区长度 (1)	报文类型 (1)	CRC8
0x55	0x0007	0x07	0x01	0xnn

数据区 (9)									
报文状态 (1)								地址 (4)	类型 (1)
B7	B6	B5	B4	B3	B2	B1	B0	目地址	Device Type
0	RC	1	1	1	1	1	1		
0x1F								TargetAddr	<a href="#">见设备类型代码</a>
									Cmd Type (1)
									0x01

可选数据区长度（7）								
报文控制类型（1）								保留（6）
TEL_CTRL								
B7	B6	B5	B4	B3	B2	B1	B0	
0	0	0	0	0	0	0	0：简单透传	0x000000000000
							1：带重发机制	

数据校验 (1)
----------

数据区	可选数据区
CRC8	

## 2.1.2 设备返回的数据格式

同步字(1)	报文头(3)			报头校验(1)
	数据区长度(2)	可选数据区长度(1)	报文类型(1)	CRC8
0x55	0x0009	0x07	0x02	0xnn

数据区(9)									
报文状态(1)					地址(4)	类型(1)	数据(3)		
B7	B6	B5	B4	B3	B2	B1	B0	目地址	Device Type
0	RC	1	1	1	1	1	0		
0x1E					SourceAddr	<a href="#">见设备类型代码</a>	Cmd Type(1)	Reportdata(2)	
							0x01	Channel(1)	Switch(1)
								0xnn	0xnn

可选数据区长度(7)	
报文控制类型(1)	保留(6)
RSSI(场强)	
0xnn	0x000000000000

数据校验(1)	
数据区	可选数据区
CRC8	

## 2.2 基本开关控制

### 2.2.1 主机发送的数据格式

同步字(1)	报文头(3)	报头校验(1)
--------	--------	---------

	数据区长度 (2)	可选数据区长度 (1)	报文类型 (1)	CRC8
0x55	0x0009	0x07	0x01	0xnn

数据区 (9)									
报文状态 (1)								地址 (4)	类型 (1)
B7	B6	B5	B4	B3	B2	B1	B0	目地址	Device Type
0	RC	1	1	1	1	1	1		
0x1F								TargetAddr	<a href="#">见设备类型代码</a>
									0x02
									0xnn
									0xnn

可选数据区长度（7）								
报文控制类型（1）								保留（6）
TEL_CTRL								
B7	B6	B5	B4	B3	B2	B1	B0	
0	0	0	0	0	0	0	0：简单透传	0x000000000000
							1：带重发机制	

数据校验（1）	
数据区	可选数据区
CRC8	

## 2.2.2 设备返回的数据格式

同步字 (1)	报文头 (3)				报头校验 (1)
	数据区长度 (2)	可选数据区长度 (1)	报文类型 (1)	CRC8	
0x55	0x0009	0x07	0x02	0xnn	

数据区 (9)									
报文状态 (1)								地址 (4)	类型 (1)
B7	B6	B5	B4	B3	B2	B1	B0	目地址	Device Type
0	RC	1	1	1	1	0	0		
0x1E								SourceAddr	<a href="#">见设备类型代码</a>
									0x02
									0xnn
									0xnn

可选数据区长度 (7)	
报文控制类型 (1)	保留 (6)
RSSI (场强)	
0xnn	0x000000000000

数据效验 (1)	
数据区	可选数据区
CRC8	

## 2.3 主动状态上报

### 2.3.1 未知事件

同步字 (1)	报文头 (3)			报头效验 (1)
	数据区长度 (2)	可选数据区长度 (1)	报文类型 (1)	CRC8
0x55	0x0009	0x07	0x03	0xnn

数据区（9）												
报文状态（1）								地址（4）	类型（1）	数据（3）		
B7	B6	B5	B4	B3	B2	B1	B0	目地址	Device Type	EventType（1）	reportData（2）	
0	RC		1	1	1	1	0				Channel（1）	Switch（1）
			0x1E		SourceAddr	<a href="#">见设备类型代码</a>				0x80	0xnn	0xnn

可选数据区长度 (7)	
报文控制类型 (1)	保留 (6)
RSSI (场强)	
0xnn	0x000000000000

数据效验 (1)	
数据区	可选数据区
CRC8	

## 2.3.2 心跳

同步字(1)	报文头(3)			报头校验(1)
	数据区长度(2)	可选数据区长度(1)	报文类型(1)	CRC8
0x55	0x0009	0x07	0x03	0xnn

数据区(9)									
报文状态(1)								地址(4)	类型(1)
B7	B6	B5	B4	B3	B2	B1	B0	目地址	Device Type
0	RC	1	1	1	1	1	0		
0x1E								SourceAddr	<a href="#">见设备类型代码</a>
									0x81
									0xnn
									0xnn

可选数据区长度（7）	
报文控制类型（1）	保留（6）
RSSI (场强)	
0xnn	0x000000000000

数据效验（1）	
数据区	可选数据区
CRC8	

## 2.3.3 按钮事件

同步字(1)	报文头(3)			报头校验(1)
	数据区长度(2)	可选数据区长度(1)	报文类型(1)	CRC8
0x55	0x0009	0x07	0x03	0xnn

数据区(9)									
报文状态(1)								地址(4)	类型(1)
B7	B6	B5	B4	B3	B2	B1	B0	目地址	Device Type
0	RC	1	1	1	1	1	0		
									0x81
									0xnn
									0xnn

		0x1E	SourceAddr	<a href="#">见设备类型代码</a>	0x82	0xnn	0xnn
--	--	------	------------	-------------------------	------	------	------

可选数据区长度（7）	
报文控制类型（1）	保留（6）
RSSI (场强)	
0xnn	0x000000000000

数据效验 (1)	
数据区	可选数据区
CRC8	

### 2.3.4 自电开关事件

同步字 (1)	报文头 (3)			报头效验 (1)
	数据区长度 (2)	可选数据区长度 (1)	报文类型 (1)	CRC8
0x55	0x0009	0x07	0x03	0xnn

数据区 (9)									
报文状态 (1)				地址 (4)	类型 (1)	数据 (3)			
B7	B6	B5	B4	B3	B2	B1	B0	目地址	reportData (2)
0	RC	1	1	1	1	0			Channel (1) Switch (1)
		0x1E			SourceAddr	<a href="#">见设备类型代码</a>	0x83	0xnn	0xnn

可选数据区长度（7）	
报文控制类型（1）	保留（6）
RSSI (场强)	
0xnn	0x000000000000

数据效验 (1)	
数据区	可选数据区
CRC8	

## 2.3.5 上电

同步字(1)	报文头 (3)			报头校验 (1)
	数据区长度 (2)	可选数据区长度 (1)	报文类型 (1)	CRC8
0x55	0x0009	0x07	0x03	0xnn

数据区（9）													
报文状态（1）								地址（4）		类型（1）		数据（3）	
B7	B6	B5	B4	B3	B2	B1	B0	目地址	Device Type	EventType（1）	reportData（2）		
0	RC	1	1	1	1	0	Channel（1）				Switch（1）		
		0x1E			SourceAddr		<a href="#">见设备类型代码</a>		0x85		0xnn	0xnn	

可选数据区长度 (7)	
报文控制类型 (1)	保留 (6)
RSSI (场强)	
0xnn	0x000000000000

数据校验 (1)	
数据区	可选数据区
CRC8	

## 2.4 管理命令

### 2.4.1 MeBox 发送管理命令

同步字(1)	报文头 (3)			报头校验 (1)
	数据区长度 (2)	可选数据区长度 (1)	报文类型 (1)	CRC8
0x55	0x0007+n	0x07	0x01	0xnn

数据区 (7+n)											
报文状态 (1)				地址 (4)	类型 (1)	数据 (1+n)					
B7	B6	B5	B4	B3	B2	B1	B0	目地址	Device Type	Mgr Type (1)	MgrData (n)
0	RC	1	1	1	0	1	1				

可选数据区长度（7）								
报文控制类型（1）								保留（6）
TEL_CTRL								
B7	B6	B5	B4	B3	B2	B1	B0	
0	0	0	0	0	0	0	0：简单透传	
							1：带重发机制	0x00000000000000

### 2.4.2 设备回复管理命令

数据区（7+n）														
报文状态（1）								地址（4）		类型（1）		数据（1+n）		
B7	B6	B5	B4	B3	B2	B1	B0	目地址	Device Type	Mgr Type（1）	MgrData（n）			
0	RC		1	1	1	0	1							
			0x1D			SourceAddr					<a href="#">见设备类型代码</a>	见管理命令	取决于具体的命令	

可选数据区长度（7）								
报文控制类型（1）								保留（6）
TEL_CTRL								
B7	B6	B5	B4	B3	B2	B1	B0	
0	0	0	0	0	0	0	0：简单透传	0x000000000000
							1：带重发机制	



数据效验 (1)	
数据区	可选数据区
CRC8	

## 五、 报文示例

下面以无挥手感应的 3 路单火线墙壁开关为例，进行报文的说明，其他类似

### 1. 下发状态查询报文

➤ 主机发送 eg) 简单透传/RC=0x10

同步字	报文头	报头效验	数据区	可选数据	数据区校验
0x55	0x00070701	0xnn	0x2F112233449F01	0x00000000000000	0xnn

➤ 设备回复 eg) 简单透传/RC=0x10/三路全开

同步字	报文头	报头效验	数据区	可选数据	数据区校验
0x55	0x00090702	0xnn	0x2E112233449F010707	0xnn00000000000000	0xnn

### 2. 下发状态控制报文

➤ 主机发送 eg) 简单透传/RC=0x10/三路全开

同步字	报文头	报头效验	数据区	可选数据	数据区校验
0x55	0x00090701	0xnn	0x2F112233449F020707	0x00000000000000	0xnn

➤ 设备回复 eg) 简单透传/RC=0x10/三路全开

同步字	报文头	报头效验	数据区	可选数据	数据区校验
0x55	0x00090702	0xnn	0x2E112233449F020707	0xnn00000000000000	0xnn

### 3. 按钮事件上报

➤ 设备上报 eg) 简单透传/RC=0x10/一路打开，其余关

同步字	报文头	报头效验	数据区	可选数据	数据区校验
0x55	0x00090703	0xnn	0x2E112233449F020701	0xnn00000000000000	0xnn

## 六、 附录 A

### 1. 设备类型代码

设备类型 (1)	
类型值	类型名
0x01	单开、双开面板 (rocker 型)
0x02	四开面板 (翻转型)
0x81	单路接收器
0x9D	无挥手单火线墙壁感应开关 1 路
0x9E	无挥手单火线墙壁感应开关 2 路
0x9F	无挥手单火线墙壁感应开关 3 路

### 2. 命令类型代码

命令类型 (1)	
类型值	类型名
0x01	状态查询
0x02	状态控制

### 3. 事件类型代码 (仅支持有源设备)

事件类型 (1)	
类型值	类型名
0x80	位置事件
0x81	心跳
0x82	按钮事件
0x83	自电开关事件
0x85	上电

### 4. 通道 (回路) 与开关

通道 (回路) (1)							
B7	B6	B5	B4	B3	B2	B1	B0
回路 8	回路 7	回路 6	回路 5	回路 4	回路 3	回路 2	回路 1
0: OFF 1: ON	0: OFF 1: ON	0: OFF 1: ON	0: OFF 1: ON	0: OFF 1: ON	0: OFF 1: ON	0: OFF 1: ON	0: OFF 1: ON

## 5. 管理命令

管理命令类型 (1)		
类型值	类型名	备注
0x01	读取固件版本号	
0x02	配置上电默认输出	Byte1_bit0-7: channel, Byte2_bit0-7: default output, 1=ON, 0=OFF;
0x03	读取上电默认输出	
0x04	配置中继功能	Byte1: 0=OFF,1=ON
0x05	读取中继功能	
0x06	配置心跳	控制/应答: Byte1: 0=OFF,1=ON(打开设备心跳)
0x07	读取心跳配置	控制: NA Byte1: 0=OFF,1=ON
0x08	读取所有或指定通道的 ID 列表长度	控制: 通道 MASK; 应答: Byte1: 通道 MASK; Byte2-5: 各通道 ID 表长度。每 4bit 表示一个通道
0x09	删除所有或指定通道的 ID 表	控制: Byte1: 通道 MASK; 应答: Byte1: 结果。0=OK
0x0A	读取所有或指定通道的完整 ID 表	控制: 通道 MASK; 应答: Byte:通道 MASK; Byte2:bit7-4:ID 表长度;bit3-0:ID 单元序号; Byte3-8:ID 单元 typedef struct { u8 cmdType; u8 cmdData; u32 chipId; }ID_ENTRY_TYPE; 每 10ms 回复一帧直到 ID 列表全部回复完
0x0B	读取指定通道的指定序号单条 ID	控制: Byte1: 通道 MASK; Byte2: bit7-4:0;bit3-0:ID 单元序号; 应答: Byte1: 通道 MASK; Byte2: bit7-4:ID 表长度;bit3-0:ID 单元序号; Byte3-8: ID 单元 typedef struct { u8 cmdType; u8 cmdData; u32 chipId; }ID_ENTRY_TYPE;

		(与 0x08 配合使用: 先读长度, 再分别一条一条的读。最终实现的效果与 0x0A 类似, 只是变成了一条一条读避免连续读的过程中中间出错。)
0x0C	按 ID 单元查询单条 ID	控制: Byte1: 通道 MASK; Byte2-7: ID 单元 应答: Byte1: 查询结果。0=存在, 1=不存在
0x0D	按 ID 单元写入单条 ID	
0x0E	按 ID 单元删除单条 ID	控制: Byte1: 通道 MASK; Byte2-7: ID 单元 应答: Byte1: 删除结果。0=OK (删除 OK), 1=不存在 (同上)

## 七、 附录 B

### 1. CRC8 效验

```
#define ProcCRCTable(crc_result,data) (CRC8Table[crc_result^data])

/* 8bit-CRC: 0x07 = x8 + x2 + x + 1 */
const unsigned char CRC8Table[] = {
0x00,0x07,0x0E,0x09,0x1C,0x1B,0x12,0x15,
    0x38,0x3F,0x36,0x31,0x24,0x23,0x2A,0x2D,
    0x70,0x77,0x7E,0x79,0x6C,0x6B,0x62,0x65,
    0x48,0x4F,0x46,0x41,0x54,0x53,0x5A,0x5D,
    0xE0,0xE7,0xEE,0xE9,0xFC,0xFB,0xF2,0xF5,
    0xD8,0xDF,0xD6,0xD1,0xC4,0xC3,0xCA,0xCD,
    0x90,0x97,0x9E,0x99,0x8C,0x8B,0x82,0x85,
    0xA8,0xAF,0xA6,0xA1,0xB4,0xB3,0xBA,0xBD,
    0xC7,0xC0,0xC9,0xCE,0xDB,0xDC,0xD5,0xD2,
    0xFF,0xF8,0xF1,0xF6,0xE3,0xE4,0xED,0xEA,
    0xB7,0xB0,0xB9,0xBE,0xAB,0xAC,0xA5,0xA2,
    0x8F,0x88,0x81,0x86,0x93,0x94,0x9D,0x9A,
    0x27,0x20,0x29,0x2E,0x3B,0x3C,0x35,0x32,
    0x1F,0x18,0x11,0x16,0x03,0x04,0x0D,0x0A,
    0x57,0x50,0x59,0x5E,0x4B,0x4C,0x45,0x42,
    0x6F,0x68,0x61,0x66,0x73,0x74,0x7D,0x7A,
    0x89,0x8E,0x87,0x80,0x95,0x92,0x9B,0x9C,
```

```
    0xB1, 0xB6, 0xBF, 0xB8, 0xAD, 0xAA, 0xA3, 0xA4,  
    0xF9, 0xFE, 0xF7, 0xF0, 0xE5, 0xE2, 0xEB, 0xEC,  
    0xC1, 0xC6, 0xCF, 0xC8, 0xDD, 0xDA, 0xD3, 0xD4,  
    0x69, 0x6E, 0x67, 0x60, 0x75, 0x72, 0x7B, 0x7C,  
    0x51, 0x56, 0x5F, 0x58, 0x4D, 0x4A, 0x43, 0x44,  
    0x19, 0x1E, 0x17, 0x10, 0x05, 0x02, 0x0B, 0x0C,  
    0x21, 0x26, 0x2F, 0x28, 0x3D, 0x3A, 0x33, 0x34,  
    0x4E, 0x49, 0x40, 0x47, 0x52, 0x55, 0x5C, 0x5B,  
    0x76, 0x71, 0x78, 0x7F, 0x6A, 0x6D, 0x64, 0x63,  
    0x3E, 0x39, 0x30, 0x37, 0x22, 0x25, 0x2C, 0x2B,  
    0x06, 0x01, 0x08, 0x0F, 0x1A, 0x1D, 0x14, 0x13,  
    0xAE, 0xA9, 0xA0, 0xA7, 0xB2, 0xB5, 0xBC, 0xBB,  
    0x96, 0x91, 0x98, 0x9F, 0x8A, 0x8D, 0x84, 0x83,  
    0xDE, 0xD9, 0xD0, 0xD7, 0xC2, 0xC5, 0xCC, 0xCB,  
    0xE6, 0xE1, 0xE8, 0xEF, 0xFA, 0xFD, 0xF4, 0xF3  
};  
  
unsigned char CRC8(unsigned char *packet, unsigned int length)  
{  
    unsigned int i = 0;  
    unsigned crc_result = 0;  
  
    for(i = 0; i < length; i++){  
        crc_result = ProcCRCTable(crc_result, packet[i]);  
    }  
  
    return crc_result;  
}
```