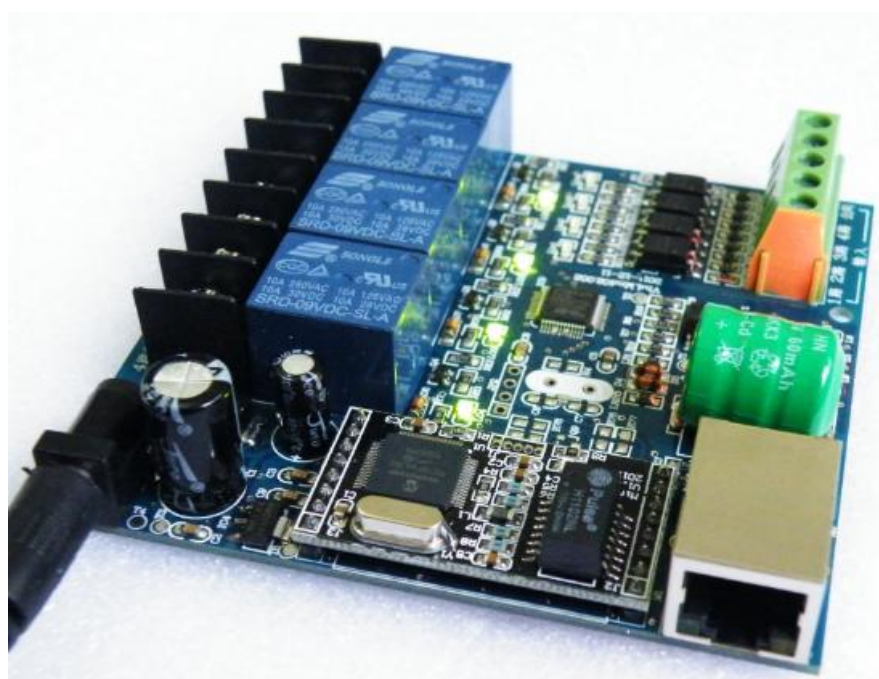


# 智能网络继电器控制板通讯协议

Version 1.2.0



文件状态: [ ] 草稿 [ ✓ ] 正式发布 [ ] 正在修改	文件标识:	MJDQ-1
	当前版本:	Ver 1.2.0
	作 者:	马相赋
	完成日期:	2010-11-12 2011-06-29 修订
	官方网站:	<a href="http://www.mndsoft.com">http://www.mndsoft.com</a>
	官方淘宝店:	<a href="http://mndsoft.taobao.com">http://mndsoft.taobao.com</a>
	联系方式:	0571-86681571 13858028385

## 1、通讯参数

RJ45 网络，支持 TCP/UP Server, Client, UDP 等通讯模式。

## 2、帧格式字节

帧是传送信息的基本单元，每帧由帧起始符标志域，控制域，数据长度域，数据域，帧信息纵向校验域及帧结束域等 6 个域组成。每个域由若干字节组成。帧格式如表格 1 所示：

表格 1 帧格式

代 码	字节数	说 明
55H	1	帧起始符
A0	4	终端逻辑地址
A1		
A2		
A3		
AAH	1	帧起始符
C	1	控制码
L	1	数据长度
DATA	变长	数据域，数据域包含标识码和用户数据
CS	1	校验码
16H	1	结束码

### 帧起始符

标识一帧信息的开始，其值为 55H=01101000B；

### 终端逻辑地址

为标识通讯的最终发起端和接收端，定义主站和终端逻辑地址。<sup>1</sup>

终端逻辑地址在通讯时用来唯一识别一个通讯的最终发起端和接收端。

地址域 A0~A4：地址域由 4 个字节构成，每字节 2 位 BCD 码。地址长度可达 12 位十进制数，可以为表号、用户号、设备号等。具体使用可由用户自行决定。当使用的地址码长度不足 4 字节时，用十六进制 00H 补足 4 字节。低地址位在先，高地址位在后。当地址为 99999999H 时，为广播地址。当地址为 AAAAAAAAAH 时，为超级地址。超级地址可以不区分设备地址进行读写操作。

### 控制码:

位	7	6	5	4	3	2	1	0
	上下行	帧错误	控制码		控制方式			

### 控制方式:

0000：本地设置，设置和读取本地的参数

0001：电平控制，按用户要求输出一个状态，直到收到用户的指令为止。

延时控制，延时到设定的时间后，自动恢复到上一个状态，

<sup>1</sup> 更好的办法是使用源地址和目的地址，地址统一编号，但是由于主站设备很少，占用 4 字节地址对于帧的利用率有较大影响，所以本规约利用传送方向标识（见控制码 C）和主站、终端地址来实现相同的功能。

方波控制，输出一个方波

控制码帧:

- 00: 读数据,
- 01: 写数据,
- 10: 保留 --- 控制
- 11: 保留

长度 (Length):

data 数据域的长度; L 为数据域的字节数, 十六进制编码, 低字节在前, 高字节在后。

校验码 (CS):

从帧起始符开始到校验码之前的所有各字节的和模 256 的余。即各字节二进制算术和, 不计超过 256 的溢出值。

结束码 (16H):

标识一帧信息的结束, 其值为 16H=00010110B

### 3、通讯数据格式:

#### 3.1、下行帧格式:

55	A0-A3	AA	C	L	ID	DATA	CS	16
----	-------	----	---	---	----	------	----	----

#### 3.2 、上行帧格式(正确返回)

55	A0-A3	AA	C	L	ID	DATA	CS	16
----	-------	----	---	---	----	------	----	----

错误返回:

位	7	6	5	4	3	2	1	0
---	---	---	---	---	---	---	---	---

#### 3.3、读控制板参数(标识码 00H 至 07H)

用于读取继电器控制板一些基本参数, 如版本号、波特率、地址、时钟等。帧格式如下:

##### 3.3.1 读取控制板有关参数

功能 : 读参数

控制码 : C=00H

数据长度 : L=01H

标识码 : I=00H 至 07H

数据示例 : 55 AA AA AA AA AA 00 01 01 A9 16

帧格式 :

起始字符 (55H)	55
终端逻辑地址	AA AA AA AA
起始字符 (AAH)	AA
控制码 C	00
数据长度 L	01

数据区 (01-07)	01
校验 (CS)	A9
帧尾 (16H)	16

数据区从 01H 开始至 07H，分别是读软件版本号（01H）、读硬件版本号（02H）、读波特率（03H）、读地址（04H）、读时钟（05H）、读延时控制单位（06H）、读方波控制单位（07H）

### 3.3.2 返回软件版本号

功能：返回软件版本号

控制码：C=80H

数据长度：L=16H

数据示例：55 12 34 56 78 AA 80 16 01 54 69 72 61 64 65 2D 73 6F 66 74 3A

56 30 31 2E 30 31 2E 30 30 FA 16

帧格式：

起始字符 (55H)	55
终端逻辑地址	12 34 56 78
起始字符 (AAH)	AA
控制码 C	80
数据长度 L	16
数据区 返回数据为十六进制 ascii 码，进行 chr(hex(x)) 还原即可，可使用 GetVersion()函数进行解码。	01 数据标识 54 69 72 61 64 65 2D 73 6F 66 74 3A 56 30 31 2E 30 31 2E 30 30
校验 (CS)	FA
帧尾 (16H)	16

返回硬件版本号和软件版本号数据一样，在此略过。

### 3.3.3 返回设备波特率

功能：返回设备波特率

控制码：C=80H

数据长度：L=02H

数据示例：55 12 34 56 78 AA 80 02 03 00 98 16

帧格式：

起始字符 (55H)	55
终端逻辑地址	12 34 56 78
起始字符 (AAH)	AA
控制码 C	80
数据长度 L	02

数据区 返回值: 00-----9600bps 01-----1200bps 02-----2400bps 03-----4800bps 04-----9600bps 05-----19200bps 06-----38400bps 07-----57600bps 08-----115200bps	03 数据标识 00
校验 (CS)	98
帧尾 (16H)	16

## 3.3.4 返回设备地址

功能 : 返回设备地址

控制码 : C=80H

数据长度 : L=05H

数据示例 : 55 12 34 56 78 AA 80 05 04 12 34 56 78 B0 16

帧格式 :

起始字符 (55H)	55
终端逻辑地址	12 34 56 78
起始字符 (AAH)	AA
控制码 C	80
数据长度 L	05
数据区 返回	04 数据标识 12 34 56 78
校验 (CS)	B0
帧尾 (16H)	16

## 3.3.5 返回设备日期、时间

功能 : 返回设备日期时间

控制码 : C=80H

数据长度 : L=08H

数据示例 : 55 12 34 56 78 AA 80 08 05 10 12 21 02 12 03 05 FF 16

帧格式 :

起始字符 (55H)	55
终端逻辑地址	12 34 56 78
起始字符 (AAH)	AA
控制码 C	80
数据长度 L	08

数据区	05 数据标识 10 年 12 月 18 日 02 周 12 时 03 分 05 秒
校验 (CS)	FF
帧尾 (16H)	16

### 3.3.6 返回延时控制单位

功能：返回延时控制单位

控制码：C=80H

数据长度：L=02H

数据示例：55 12 34 56 78 AA 80 02 06 00 9B 16

帧格式：

起始字符 (55H)	55
终端逻辑地址	12 34 56 78
起始字符 (AAH)	AA
控制码 C	80
数据长度 L	02
数据区 返回 00: 秒, 01: 分; 02: 10 分钟 03: 30 分钟	06 数据标识 00
校验 (CS)	9B
帧尾 (16H)	16

### 3.3.7 返回方波控制单位

返回方波控制单位同延时控制单位结果一样，在此略过。

### 3.3.7 代码说明

在示例源代码中过程是 cmdReadSet\_Click(), 代码如下:

```
' 循环读取 7 个项目
```

```
For i = 1 To 7
```

```
    If i > 3 Then ' 前 3 个暂时不处理
```

```
        ' 此处调用 SendEncode 协议编码函数, 生成编码并发送到端口, 有关 SendEncode
```

函数说明请参阅《编码解码函数说明》

```
        Call SendEncode(, "00", , Format$(i, "00"), True)
```

```
        Pause 1 ' 延时 1 秒
```

```
    End If
```

```
Next
```

返回数据: 在示例代码中的 MSComm\_OnComm 事件中, 具体可以参考代码注释。

3.2、读每路继电器定时时间(标识码 10H 至 1BH, 为第一路继电器至第十二路继电器序号)  
用于读取继电器的定时时间, 定时格式为每周, 每天 XX 点 XX 分 XX 秒至 XX 点 XX 分 XX 秒,  
支持离线脱离电脑定时。

注: 该功能只要设置继电器状态为定时 F8H 时才有效。

功能 : 读每路继电器定时时间

控制码 : C=00H

数据长度 : L=01H

标识码 : I=10H 至 1BH

数据示例 : 55 AA AA AA AA AA 00 01 10 B8 16

帧格式 :

起始字符 ( 55H )	55
终端逻辑地址	AA AA AA AA
起始字符 ( AAH )	AA
控制码 C	00
数据长度 L	01
数据区 (10-1B)	12
校验 ( CS )	BA
帧尾 ( 16H )	16

数据区从 10H 开始至 1BH, 分别是读取继电器第一路至第十二路定时时间。

返回应答:

控制码 : C=80H

数据长度 : L=07H

数据示例 : 55 12 34 56 78 AA 80 07 10 13 55 30 14 37 26 B3 16

帧格式 :

起始字符（55H）	55	
终端逻辑地址	00 00 00 00	
起始字符（AAH）	AA	
控制码 C	80	
数据长度 L	08	
数据区	12	数据标识，第几路
	FC	每周时间选择
	09	开启：时
	15	开启：分
	00	开启：秒
	09	关闭：时
	16	关闭：分
	00	关闭：秒
校验（CS）	D2	
帧尾（16H）	16	

其中数据区中的 FC 为周一至周日位选择标识, 需要转换为二进制位表示, FC=11111100, 顺

序从 0 至 6，代表从周一至周日选择状态，第 7 位保留。1 表示选择，0 表示不选择。

代码说明：在示例源代码中过程是 cmdTimeCheck\_Click(), 代码如下：

' 循环读取

For i = 0 To 11

    发送十六进制序号

        Call SendEncode(, "00", , Right("0" & Hex(i + 16), 2), True)

        Pause 1

Next

返回数据：在示例代码中的 MSComm\_OnComm 事件中, 具体可以参考代码注释。

3.3、读每路继电器状态(标识码 00H 至 0BH, 为第一路继电器至第十二路继电器序号, 0FH 为一次性返回全部继电器状态)

用于读取每路继电器当前状态，如开关状态、定时状态。

功能      : 读每路继电器状态

控制码    : C=00H

数据长度  : L=01H

标识码    : I=00H 至 0BH

数据示例  : 55 AA AA AA AA AA 01 01 00 A9 16

帧格式    :

起始字符 (55H)	55
终端逻辑地址	AA AA AA AA
起始字符 (AAH)	AA
控制码 C	01
数据长度 L	01
数据区 (00-0B)	00
校验 (CS)	A9
帧尾 (16H)	16

数据区从 00H 开始至 0BH，分别是读取继电器第一路至第十二路开关当前状态。

返回应答：

控制码    : C=81H

数据长度  : L=03H

数据标识  : 00H 至 0BH 读取全部 0FH

数据示例  : 55 12 34 56 78 AA 81 03 00 80 00 17 16

帧格式    :

起始字符 (55H)	55
终端逻辑地址	12 34 56 78
起始字符 (AAH)	AA
控制码 C	81
数据长度 L	03



数据区	00 数据标识, 第几路
00H: 表示关闭状态	80 处于开状态
80H: 表示开启状态	00
F8H: 表示定时状态	
校验 (CS)	17
帧尾 (16H)	16

代码说明: 在示例源代码中过程是 cmdOneStateCheck\_Click 过程。

### 3.4、读全部继电器状态(标识码 0FH, 一次性返回全部继电器开关状态)

用于读取每路继电器当前状态, 如开关状态、定时状态、延时状态。

功能 : 读全部继电器状态

控制码 : C=01H

数据长度 : L=01H

标识码 : I=0FH

数据示例 : 55 AA AA AA AA 01 01 0F B8 16

帧格式 :

起始字符 (55H)	55
终端逻辑地址	AA AA AA AA
起始字符 (AAH)	AA
控制码 C	01
数据长度 L	01
数据区	0F
校验 (CS)	B8
帧尾 (16H)	16

返回应答:

控制码 : C=81H

数据长度 : L=11H

数据标识 : 0FH

数据示例 : 55 00 00 00 00 AA 81 11 0F 02 00 00 00 00 00 00 F8 00 00 00 00

00 00 00 00 9A 16

帧格式 :

起始字符 (55H)	55
终端逻辑地址	00 00 00 00
起始字符 (AAH)	AA
控制码 C	81
数据长度 L	11
数据区	0F 数据标识, 全部状态
00H: 表示关闭状态	02 00 状态位, 需转为二进制
80H: 表示开启状态	010000000000, 每一位代表一路继
F8H: 表示定时状态	电器, 0 表示关闭, 1 表示开启状态
	00 00 分别表示延时和方波控

	制的单位
	00 00 00 F8 00 00 00 00 00 00 00 00 上面 12 个字节分别代表每一路继电器的其他状态，见左边说明。
校 验 （ CS ）	9A
帧 尾 （ 16H ）	16

代码说明：在示例源代码中过程是 cmdMul ti StateCheck\_Click()过程。

### 3.5、写控制板参数

用于设置继电器控制板一些基本参数，如波特率、地址、时钟、定时开关、定时时间等。帧格式如下：

#### 3.5.1 写波特率

功能       ： 设定继电器控制板波特率  
控制码     ： C=10H  
数据长度   ： L=02H  
标识码     ： 02H  
数据示例   ： 55 AA AA AA AA 10 02 03 00 9B 16  
帧格式     ：

起 始 字 符 （ 55H ）	55
终 端 逻 辑 地 址	AA AA AA AA
起 始 字 符 （ AAH ）	AA
控 制 码 C	10
数 据 长 度 L	02
数 据 区	03 数据标识
00-----9600bps	00
01-----1200bps	
02-----2400bps	
03-----4800bps	
04-----9600bps	
05-----19200bps	
06-----38400bps	
07-----57600bps	
08-----115200bps	
校 验 （ CS ）	9B
帧 尾 （ 16H ）	16

设定波特率缺省值为 9600bps, 写入完毕，关闭串口更换刚才设定的波特率进行测试，正常读取数据则说明设置成功。

#### 3.5.2 写设备地址

功能       ： 设定继电器控制板逻辑地址，编号 4 字节，自定义  
控制码     ： C=10H

数据长度：L=05H

标识码：04H

数据示例：55 AA AA AA AA 10 05 04 12 34 56 78 D4 16

帧格式：

起始字符（55H）	55
终端逻辑地址	AA AA AA AA
起始字符（AAH）	AA
控制码 C	10
数据长度 L	05
数据区	04 数据标识 12 34 56 78
校验（CS）	D4
帧尾（16H）	16

返回应答：

功能：返回设备地址

控制码：C=90H

数据长度：L=16H

数据示例：55 12 34 56 78 AA 90 01 04 A8 16

帧格式：

起始字符（55H）	55
终端逻辑地址	12 34 56 78
起始字符（AAH）	AA
控制码 C	90
数据长度 L	01
数据区	04 数据标识
校验（CS）	A8
帧尾（16H）	16

### 3.5.3 写设备日期、时钟

功能：设定继电器控制板日期和时间

控制码：C=10H

数据长度：L=08H

标识码：05H

数据示例：55 AA AA AA AA AA 10 08 05 10 12 18 06 09 25 40 72 16

帧格式：

起始字符（55H）	55
终端逻辑地址	AA AA AA AA
起始字符（AAH）	AA
控制码 C	10
数据长度 L	08
数据区	05 数据标识 10 年

	12 月
	18 日
	06 周
	09 时
	25 分
	40 秒
校 验 (CS)	72
帧 尾 (16H)	16

返回应答: 同上

### 3.5.4 写设备延时控制单位

功能 : 设定设备延时控制单位

控制码 : C=10H

数据长度 : L=02H

标识码 : 06H

数据示例 : 55 AA AA AA AA 10 02 06 00 BF 16

帧格式 :

起 始 字 符 (55H)	55
终 端 逻 辑 地 址	AA AA AA AA
起 始 字 符 (AAH)	AA
控 制 码 C	10
数 据 长 度 L	02
数 据 区	06 数据标识
00: 秒	00
01: 分	
02: 10 分钟	
03: 30 分钟	
校 验 (CS)	BF
帧 尾 (16H)	16

返回应答: 同上

### 3.5.5 写方波控制单位参见设备延时控制单位

示例代码说明: 以上写参数代码示例 参见 cmdWriteSet\_Click() 过程中。

### 3.5.6 写定时开启、关闭状态

功能 : 设定继电器定时开启、关闭状态

控制码 : C=11H

数据长度 : L=02H

标识码 : 01H 至 0BH

数据示例 : 55 AA AA AA AA 11 02 00 F8 B2 16

帧格式 :

起始字符 (55H)	55
终端逻辑地址	AA AA AA AA
起始字符 (AAH)	AA
控制码 C	11
数据长度 L	02
数据区	00 数据标识
00 取消定时 10H至1BH(1-12路)	F8
F8 执行定时 01H至0BH(1-12路)	
校验 (CS)	B2
帧尾 (16H)	16

返回应答： 见返回应答格式,取消定时 控制码返回 91H。  
 示例代码说明：以上码示例 参见 cmdTimeSwitch\_Click()过程。

### 3.5.7 写定时时间

功能： 设定继电器定时开启和定时关闭时间  
 控制码： C=10H  
 数据长度： L=08  
 标识码： 10H至1BH  
 数据示例： 55 AA AA AA AA AA 10 08 10 F8 14 15 00 14 20 00 24 16  
 帧格式：

起始字符 (55H)	55
终端逻辑地址	AA AA AA AA
起始字符 (AAH)	AA
控制码 C	10
数据长度 L	08
数据区 10H至1BH(1-12路)	10 数据标识
	F8 周一至周日选择标识，转换为二进制为 11111000，顺序从 0 至 6，代表周一至周日，第 7 位保留。 14 15 00 开启 时/分/秒 14 20 00 关闭 时/分/秒
校验 (CS)	24
帧尾 (16H)	16

返回应答： 见返回应答格式

## 3.6、控制

控制主要用于立即控制继电器开关及定时控制、分组选择控制、翻转、方波输出、延时控制等对继电器的操作。

### 3.6.1、立即开启/立即关闭

功能 : 立即执行哪一路继电器开启或者关闭  
 控制码 : C=11H  
 数据长度 : L=07H  
 标识码 : 开启标识 00H 至 0BH 关闭标识 10H 至 1BH  
 数据示例 : 55 AA AA AA AA AA 11 02 00 00 BA 16  
 帧格式 :

起始字符 ( 55H )	55
终端逻辑地址	AA AA AA AA
起始字符 ( AAH )	AA
控制码 C	11
数据长度 L	02
数据区 开启标识 00H 至 0BH(1-12 路) 关闭标识 10H 至 1BH(1-12 路)	00 数据标识 00
校验 ( CS )	BA
帧尾 ( 16H )	16

返回应答:

控制码 : C=开启返回 90H, 关闭返回 91H  
 数据长度 : L=02H  
 数据示例 : 55 12 34 56 78 AA 90 02 00 80 25 16  
 帧格式 :

起始字符 ( 55H )	55
终端逻辑地址	12 34 56 78
起始字符 ( AAH )	AA
控制码 C	90
数据长度 L	02
数据区 数据标识: 开 (00 至 0B, 第 1 至 12 路) 数据标识: 关 (10 至 1B, 第 1 至 12 路) 数据 80H 表示开 00H 表示关	00 数据标识, 表示第一路 80
校验 ( CS )	25
帧尾 ( 16H )	16

示例代码说明: 以上码示例 参见 chkSwitch\_Click 过程。

### 3.6.2、全部开启/全部关闭

功能 : 执行全部继电器开启或者全部关闭  
 控制码 : C=11H

数据长度：L=0DH

标识码：全部开启标识 0FH 全部关闭标识 1FH

数据示例：55 AA AA AA AA 11 0D 0F 00 00 00 00 FF FF FF FF FF FF FF

CC 16

帧格式：

起始字符（55H）	55
终端逻辑地址	AA AA AA AA
起始字符（AAH）	AA
控制码 C	11
数据长度 L	0D
数据区 全部开启标识 0FH 全部关闭标识 1FH 00 为要控制 FF 为不处理 该示例为控制板缺省 4 路演示。	0F 数据标识 00 第一路 00 第二路 00 . 00 . FF . FF . FF . FF . FF . FF . FF 第十二路
校验（CS）	CC
帧尾（16H）	16

返回应答：

控制码：C=90H（开启返回） 91H（关闭返回）

数据长度：L=03H

数据示例：55 12 34 56 78 AA 90 03 0F 0F 00 C4 16

帧格式：

起始字符（55H）	55
终端逻辑地址	12 34 56 78
起始字符（AAH）	AA
控制码 C 90H（开启返回） 91H（关闭返回）	90
数据长度 L	03
数据区 数据标识 0F 开 1F 关	0F 数据标识 0F 00
校验（CS）	C4
帧尾（16H）	16

示例代码说明：以上码示例 参见 cmdSwAll\_Click 过程。

### 3.6.3、有选择开启/有选择关闭

功能：分别指定哪几路继电器执行开启或者关闭

控制码：C=11H

数据长度：L=0DH

标识码：开启标识 0FH 关闭标识 1FH

数据示例：55 AA AA AA AA AA 11 0D 0F 00 FF 00 FF FF FF FF FF FF FF FF

CA 16

帧格式：

起始字符（55H）	55
终端逻辑地址	AA AA AA AA
起始字符（AAH）	AA
控制码 C	11
数据长度 L	0D
数据区 全部开启标识 0FH 全部关闭标识 1FH 00 为控制开 FF 为控制关 该示例为控制板缺省 4 路 演示有选择开启第一路和第三路。	0F 数据标识 00 第一路（开启） FF 第二路 00 第三路（开启） FF 第四路 FF . FF . FF . FF . FF . FF . FF . FF 第十二路
校验（CS）	CA
帧尾（16H）	16

返回应答：见 3.5.2、全部开启/全部关闭返回应答格式

示例代码说明：以上码示例 参见 cmdGroupSwitch\_Click 过程。

### 3.6.4、翻转

功能：对应切换开关状态。

控制码：C=11H

数据长度：L=0DH

标识码：2FH

数据示例：55 AA AA AA AA AA 11 0D 2F 00 00 00 00 FF FF FF FF FF FF FF FF

EC 16

帧格式：

起始字符（55H）	55
终端逻辑地址	AA AA AA AA
起始字符（AAH）	AA
控制码 C	11



数据长度 L	0D
数据区	2F 数据标识
全部开启标识 0FH	00 第一路 (开启)
全部关闭标识 1FH	00 第二路
00 为控制开	00 第三路 (开启)
FF 为控制关	00 第四路
该示例为控制板缺省 4 路	FF .
演示有选择开启第一路和第三路。	FF .
	FF .
	FF .
	FF .
	FF .
	FF .
	FF 第十二路
校验 (CS)	EC
帧尾 (16H)	16

返回应答:

控制码 : C=92H

数据长度 : L=03H

数据示例 : 55 12 34 56 78 AA 92 03 2F 00 00 D7 16

帧格式 :

起始字符 (55H)	55
终端逻辑地址	12 34 56 78
起始字符 (AAH)	AA
控制码 C	92
数据长度 L	03
数据区	2F 数据标识
数据标识 0F 翻转	00
00 翻转关闭	00
校验 (CS)	D7
帧尾 (16H)	16

示例代码说明: 以上码示例 参见 cmdChange\_Click 过程。

### 3.6.5、方波输出

功能 : 指定哪一路继电器按一定时间重复开关

控制码 : C=11H

数据长度 : L=0DH

标识码 : 2FH

数据示例 : 55 AA AA AA AA AA 11 0D 2F 01 00 02 05 FF FF FF FF FF FF FF F4 16

帧格式 :

起始字符 (55H)	55
------------	----

终端逻辑地址	AA AA AA AA
起始字符 (AAH)	AA
控制码 C	11
数据长度 L	0D
数据区 全部开启标识 0FH 全部关闭标识 1FH 00 为要控制 FF 为不处理 该示例为控制板缺省 4 路演示。	2F 数据标识 01 第一路 00 第二路 02 . 05 . FF . FF . FF . FF . FF . FF . FF 第十二路
校验 (CS)	F4
帧尾 (16H)	16

返回应答： 见 3.5.2、全部开启/全部关闭返回应答格式

示例代码说明：以上码示例 参见 cmdSquare\_Click 过程。

### 3.6.6、延时 n 秒开启/延时 n 秒关闭

功能：指定哪一路继电器延时 n 秒开或者关

控制码：C=11H

数据长度：L=0DH

标识码：0FH

数据示例：55 AA AA AA AA AA 11 0D 0F 05 08 FF FF FF FF FF FF FF FF FF D7 16

帧格式：

起始字符 (55H)	55
终端逻辑地址	AA AA AA AA
起始字符 (AAH)	AA
控制码 C	11
数据长度 L	0D
数据区 开启标识 0FH 关闭标识 1FH 05 为延时秒数 FF 为不处理 该示例为控制板缺省 4 路演示。	0F 数据标识 05 第一路延时 5 秒 08 第二路延时 8 秒 FF . FF . FF . FF . FF . FF .

	FF . FF . FF . FF 第十二路
校 验 (CS)	D7
帧 尾 (16H)	16

返回应答： 见 3.5.2、全部开启/全部关闭返回应答格式

示例代码说明：以上码示例 参见 cmdDelaySw\_Click 过程。

### 3.6.7、关闭延时 n 秒开启/开启延时 n 秒关闭

功能：指定哪一路继电器关闭延时 n 秒后重新开启或开启延时 n 秒再关闭

控制码：C=11H

数据长度：L=02H

标识码：关闭 00H-0BH 开：10H-1BH

数据示例：55 AA AA AA AA 11 02 00 05 BF 16

帧格式：

起始字符 (55H)	55
终端逻辑地址	AA AA AA AA
起始字符 (AAH)	AA
控制码 C	11
数据长度 L	02
数据区 关闭延时 n 秒开启：00-0B 即第 1 路至第 12 路 开启延时 n 秒关闭：10-1B 即第 1 至第 12 路	00 数据标识，表示第一路 05 延时 5 秒
校 验 (CS)	BF
帧 尾 (16H)	16

返回应答： 见 3.6.2、全部开启/全部关闭返回应答格式

示例代码说明：以上码示例 参见 cmdDelayDemo\_Click 过程。

## 4、示例代码解码编码函数

打开随控制板附带的 VB 源代码示例 prj SwtchDemo.vbp, 在 mdlMain 模块中有 SendEncode() 和 ReceiveDecode() 2 个函数, SendEncode() 为编码函数, ReceiveDecode() 为解码函数。

以下讲解 2 个函数入口参数格式:

### 4.1、SendEncode 函数:

\*\*\*\*\*

\*\*\*函数名: SendEncode

\*\*\*输入: StrAddress(String) -设备地址, 不指定默认超级地址 AA AA AA AA

\*\*\*: ControlCode(String) -控制码 即指定功能, 如读取、写、控制等

\*\*\*: Udata(String) -数据区, 标识码及进数据区

```

' **      : isOut          -是否向端口发送数据 , 主要为了调试时检测数据
' **输出   出: (String) -
' **功能描述: 继电器控制协议编码函数
' ****

```

isOut 参数指定是否输出到端口, 在调试时可以在 VB 的【立即窗口】使用时为 False 时则直接输出编码的协议帧到立即窗口。为 True 时则写入端口, 在立即窗口中无效并报错。

```

Public Function SendEncode(Optional StrAddress As String, Optional ControlCode As
String, Optional sFlag As String, Optional Udata As String, Optional isOut As Boolean)
As String

```

使用示例:

使用超级地址, 并输出到端口:

```
Call SendEncode(, "11", , "0200", True)
```

使用实际地址, 并输出到端口:

```
Call SendEncode("12345678", "11", "05", "0200", True)
```

#### 4.2、ReceiveDecode 函数:

```

' ****
' **函数名: ReceiveDecode
' **输入: ReceiveData(String) - 对接收到的数据进行解析
' **输出: (String) -ID+sCtrl+sFlag+sUdata 设备 ID, 控制字, 数据标识, 数据
' **功能描述: 继电器控制协议解码函数
' ****

```

该函数输出时分别输出设备地址、控制码、数据标识、数据区, 并分别用逗号 “,” 分隔, 解析数据时, 可以使用 Split(Trim\$(ReceiveData), ",") 函数赋值到字符数组进行处理。

```
Public Function ReceiveDecode(ReceiveData As String) As String
```

#### 5、数据项编码与格式

控制方式 (控制码)	标识码	数据格式	长度	码制	功能		名称
					读	写	
00	00	ASCII			*		(默认 9600 波特率) 复位时返回串口设置信息、通信地址。
	01				*		软件版本号
	02				*		硬件版本号
	03	NN			*	*	设置波特率
	04	BBBBBBBB	4		*	*	设置地址
	05	YYMMDDWWHHMMSS	7		*	*	设置日期、星期、时钟
	06	BB			*	*	延时控制的单位 (0: 秒, 1: 分; 2: 10 分钟 3: 30 分钟)

	07	BB			*	*	方波控制的单位 (0: 秒, 1: 分; 2: 10 分钟 3: 30 分钟)
	10-1B	FWEEK HHMMSS HHMMSS	7	BCD	*	*	Fweek: 周一至周日选择标志位, 参见附录 设置 1 至 12 路定时开启、关闭时间 (时分秒) 24:00:00 为无效时间。
	1F						
01	00-0B	NN	1	HEX	*	*	第 01-12 路开启控制; 数据: 00 开启后不再关闭, (需要用户命令关闭) 01-F0 关闭并延时 (NN*延时单位) 后开启; F8: 设置为定时控制。
	0F	全	12			*	FF: 无效数据 (只用于标识码 0F)
	10-1B	NN			*	*	第 01-12 路关闭控制; 数据: 00 关闭后不再开启, (需要用户命令开启) 01-F0 开启并延时 (01-FE*延时单位) 后关闭; F8: 设置为定时控制。
	1F					*	FF: 无效数据 (只用于标识码 1F)
	20-2B	NN			*	*	第 01-12 路关闭控制; 数据: 00 立即翻转。 01-F0 反复延时 (01-FE*延时单位) 后取反输出方波;
	2F					*	FF: 无效数据 (只用于标识码 2F)
	0F 1F 2F				*		各线路当前状态 (2) 延时控制的单位 (1) 方波控制的单位 (1) 延时时间标识 (12)

注: 读取单路状态时, 00 为低, 80 为高;

附录:

周一至周日定时选择标志字:

位	7	6	5	4	3	2	1	0
	保留	周六	周五	周四	周三	周二	周一	周日

错误字:

位	7	6	5	4	3	2	1	0
	保留					数据非法	DI 码错	控制码错