# REL4 V1.0 测试文档

## 1.设备说明

1.1硬件

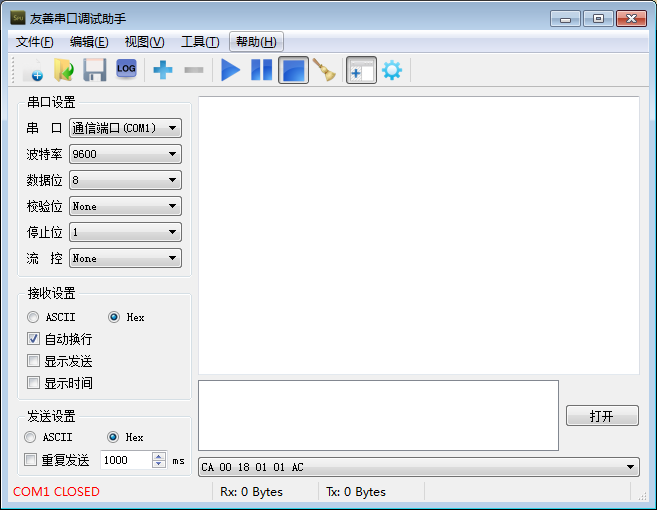
REL4设备一台，USB转232串口转换器一条，跳线若干，可联网的电脑一台，zigbee主机板一块，zigbee主板电源线一条，网线一条



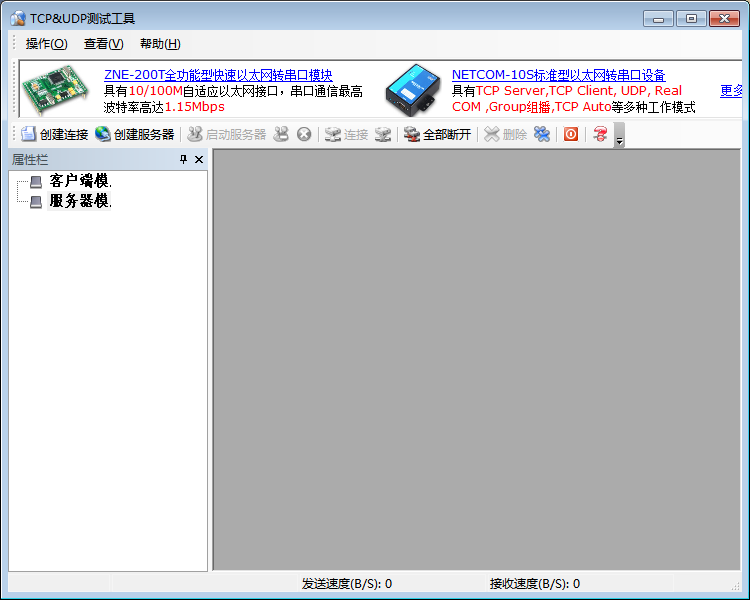
USB转RS232串口转换器

1.2软件

串口调试助手（此处使用友善串口调试助手），TCP/IP 测试工具（此处使用TCP&UDP测试工具）



友善串口调试助手



TCP&UDP测试工具

## 2．连线方式以及初始设置

连线如下所示：

使用RS232连线发送数据时，232的Tx连在RXD处，232的Rx连在TXD处，232的GND连在GND处。

和常见的DB9接头的对应方式见下图。可根据该图进行电缆焊接。连接方式为DB9接头的RxD、TxD分别接232输入串口的TxD和RxD，DB9的GND与232输入串口的GND相连。

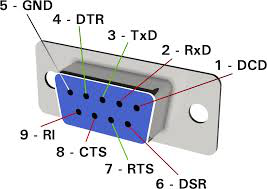
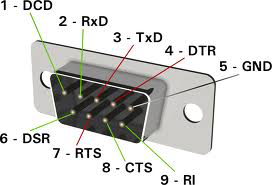
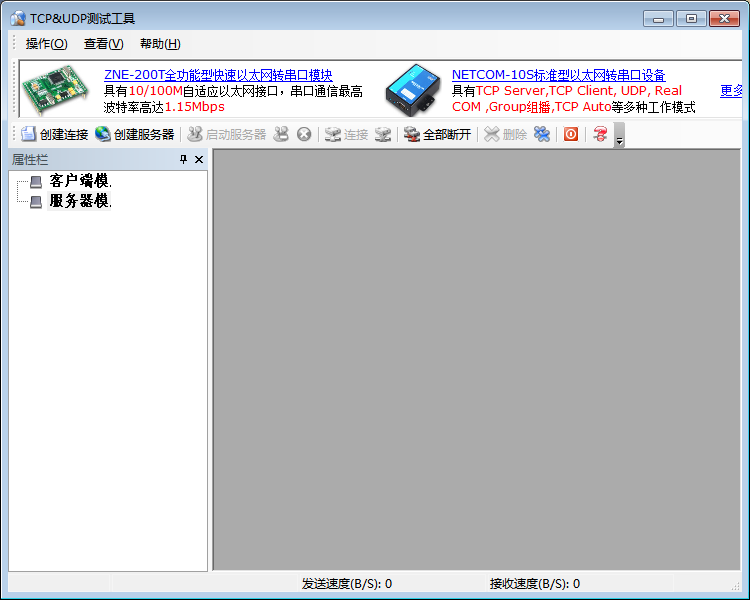
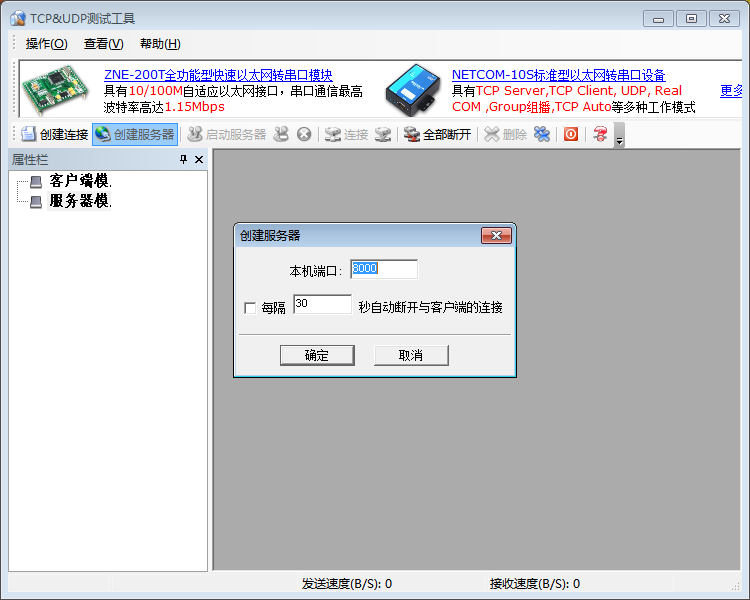


图5 DB9接头端口定义

232输入串口波特率为9600bps，8位数据，1位停止位，没有校验位。

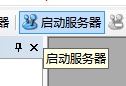
Zigbee主机连接网线和电源线即可，看到LAN口指示灯闪烁即正常工作。

联网的电脑设置IP为192.168.1.1。打开TCP&UDP测试工具软件点击创建服务器

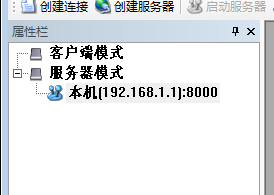
本机端口设置为8000，确定



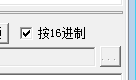
点击启动服务器



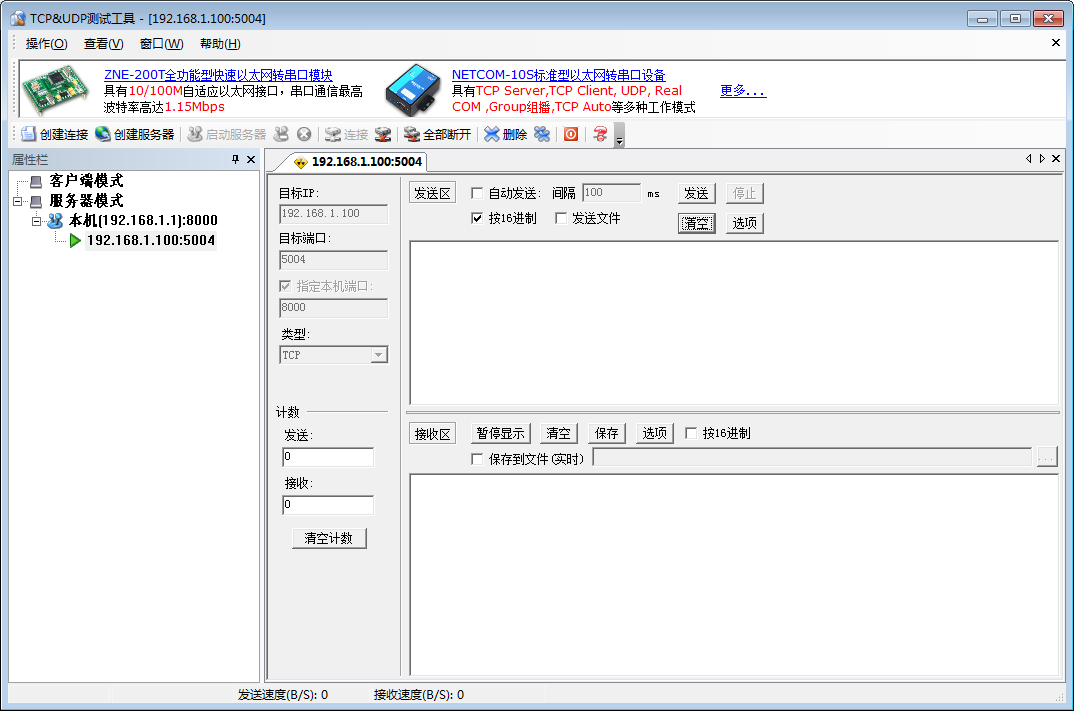
此时变为如下状态：



与EXB-CIR用网线相连接（若直接相连则立即生效，若使用交换机相连，则需等待30秒），设备与电脑的TCP服务器连接上，点击按16进制



完成电脑端TCP服务器配置，如下图所示



打开友善串口调试助手，按照USB转RS232串口转换器接入的电脑端的串口号进行设置，如下图（此处连接的为电脑端的COM4口）：

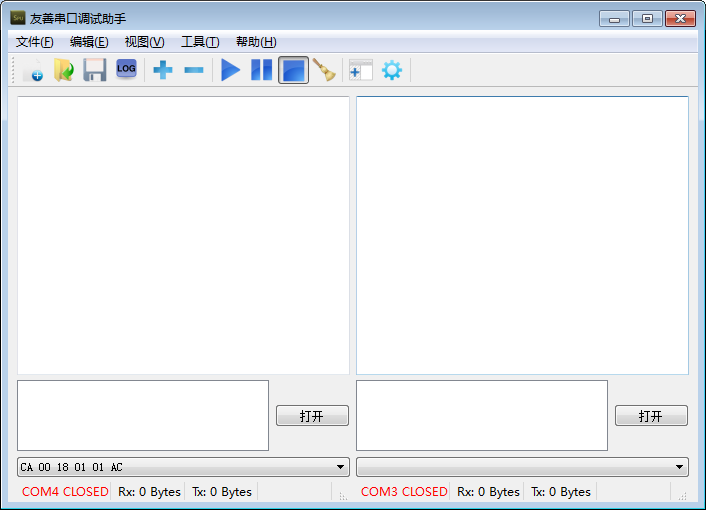


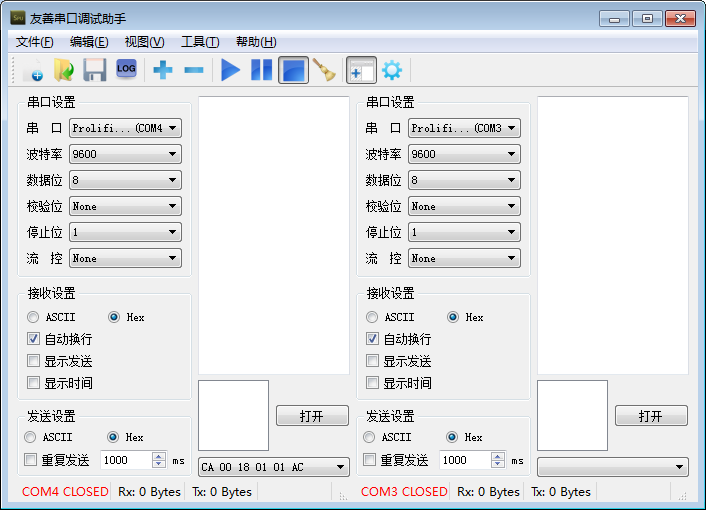
要增加同时观察的串口，按“+”

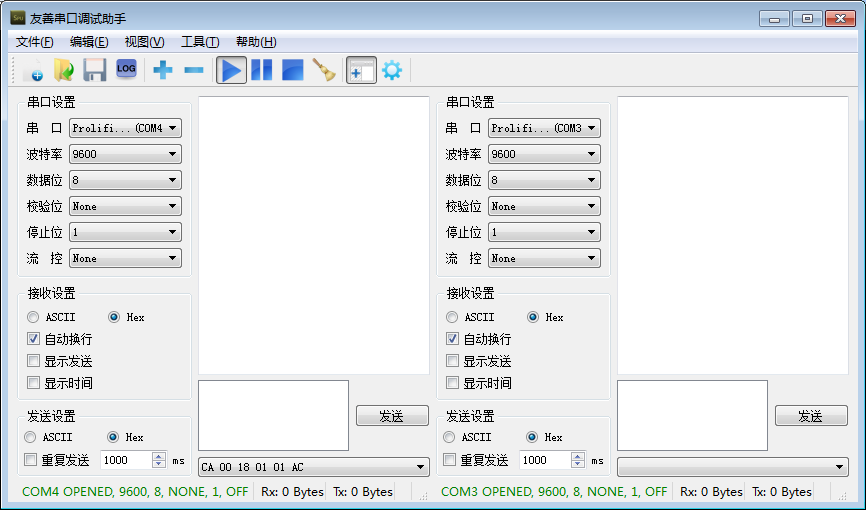


弹出如下窗口（进行 配置如下图，串口对应实际）



点击OK

按C:\Users\Administrator\Desktop\2.png打开配置面板，如下设置

按C:\Users\Administrator\Desktop\QQ截图20160310101826.png打开串口，如下所示完成串口调试助手配置：

Zigbee主机和REL4设备全部配置好，上电初始化之后，zigbee主机的2个LED等均亮，LED2间隔性闪烁一下，REL4的3个LED最左一个闪烁，中间间歇性闪烁一下，右边的一个常亮，即工作正常。

Zigbee模块默认的POLL\_RATE较长，无法进行即时回码，所以先要对POLL\_RATE进行设置，之后才能进行接下来的测试工作。

## 3.通讯协议

### 3.1 232串口通讯协议

#### 3.1.1

设备可以通过指令改变4路继电器的开关状态，通讯协议如下表1：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 起始符 | 长度 | 功能选择 | 设备地址 | 8路继电器状态 |
| AA 55（16进制） | 0A | A7 | 00 | d1 d2 d3 d4 XX XX XX XX |

表1 改变4路继电器状态命令格式

4路继电器状态设置中,d1~d4分别对应4路继电器开关位（均为16进制，之后的表中也相同），00为关，非0为开，0xFF为当前继电器跳过不进行改变。设置成功串口会返回数据AA 55 03 7A 00 CC。

#### 3.1.2

设备可以通过指令改变上电时4路继电器的初始状态，通讯协议如下表2：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 起始符 | 长度 | 功能选择 | 设备地址 | 4路继电器状态 |
| AA 55（16进制） | 0A | A2 | 00 | d1 d2 d3 d4 XX XX XX XX |

表2 改变4路继电器上电时初始状态命令格式

设置成功后串口会返回数据AA 55 03 2A 00 CC。设置完成后，每次设备重新上电4路继电器都会初始化为设定的状态。

#### 3.1.3

设备可以通过指令查询当前4路继电器的开关状态，通讯协议为：AA 55 02 C3 00。串口会返回当前4路继电器的状态，返回数据通讯协议如下表3：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 起始符 | 长度 | 功能选择 | 设备地址 | 4路继电器状态 |
| AA 55（16进制） | 0A | 3C | 00 | d1 d2 d3 d4 XX XX XX XX |

表3 查询指令返回数据命令格式

d1~d4分别代表当前4路继电器的状态，00为关，01为开。

#### 3.1.4

为方便使用，设备可以进行多个场景的保存和直接读取并设置为场景对应状态，设置设备场景状态的通讯协议如下表4：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 起始符 | 长度 | 功能选择 | 设备地址 | 场景编号位 | 4路继电器状态 |
| AA 55（16进制） | 0B | A5 | 00 | sceNo | d1 d2 d3 d4 XX XX XX XX |

表4 设置设备场景命令格式

设置成功串口会收到返回的数据：AA 55 03 5A 00 sceNo。场景编号位取值范围为sceNo < 85。

设置场景成功后，可以直接调用场景号来控制继电器的状态，调用场景号的通讯协议如下表5：

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 起始符 | 长度 | 功能选择 | 设备地址 | 场景编号位 | 间隔 | 结束符 |
| AA 55（16进制） | 05 | D3 | 00 | sceNo | XX | 0x00 |

表5 调用设备场景命令格式

场景编号即为之前设置过的场景号，间隔位只起间隔作用，一般为0xFF。使用此指令，调用场景号，使4路继电器变为之前设置的状态。

#### 3.1.5

设备具有广播功能，可以广播所有设备控制继电器状态，也可以广播所有设备调用场景，即使用广播模式时无需设置设备地址。

广播设置所有设备继电器状态的通讯协议如下表6：

|  |  |  |  |
| --- | --- | --- | --- |
| 起始符 | 长度 | 功能选择 | 4路继电器状态 |
| AA 55（16进制） | 09 | D4 | d1 d2 d3 d4 XX XX XX XX |

表6 广播控制继电器状态命令格式

广播调用设备场景号的通讯协议如下表7：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 起始符 | 长度 | 功能选择 | 间隔符 | 场景编号位 | 结束符 |
| AA 55（16进制） | 05 | D2 | XX XX | sceNo | XX |

表7 广播调用设备场景命令格式

间隔符只起间隔作用，一般为0xFF 0xFF，结束符一般也为0xFF。

广播功能只需输入控制命令，则所有AVB-DIN-REL4设备都会对指令进行反应。

#### 3.1.6

可以通过232串口对于zigbee进行设置，包括PANID，BAUD\_RATE，UNI\_SEC\_ADDR，POLL\_RATE，RESET功能。设置完成会自动对zigbee进行重启，无需重新上电或再次使用zigbee重启指令。

PANID（私网号）设置命令如下：

|  |  |  |  |
| --- | --- | --- | --- |
| 起始符 | 长度 | 功能选择 | PANID的值 |
| AA 55（16进制） | 07 | 33 00 01 | 00 0X 0Y 0Z |

为便于设置无需转化为16进制，PANID的值直接使用10进制进行设置，XYZ就是PANID设置成的值。例如 AA 55 07 33 00 01 00 01 00 00指令即设置zigbee的PANID为100。

BAUD\_RATE（波特率）设置命令如下：

|  |  |  |  |
| --- | --- | --- | --- |
| 起始符 | 长度 | 功能选择 | BAUD\_RATE的值 |
| AA 55（16进制） | 07 | 33 00 02 | XX XX XX XX |

BAUD\_RATE设置使用16进制，四位BAUD\_RATE即为16进制的波特率的值，例如AA 55 07 33 00 02 00 01 c2 00指令即为设置zigbee波特率为00 01 c2 00（16进制，转化为10进制为115200）。

UNI\_SEC\_ADDR（zigbee地址）设置命令如下：

|  |  |  |  |
| --- | --- | --- | --- |
| 起始符 | 长度 | 功能选择 | UNI\_SEC\_ADDR的值 |
| AA 55（16进制） | 07 | 33 00 03 | 00 00 00 0X |

因为zigbee主机最多同时连接3个终端设备，所以0X的 值为 01~03，例如AA 55 07 33 00 03 00 00 00 01指令即为设置zigbee的设备地址为01。

POLL\_RATE（查询网络包间隔时间）设置命令如下：

|  |  |  |  |
| --- | --- | --- | --- |
| 起始符 | 长度 | 功能选择 | POLL\_RATE的值 |
| AA 55（16进制） | 07 | 33 00 04 | 00 00 0X 0Y |

POLL\_RATE为终端查询主机端数据包的间隔时间，若时间较长，则无法及时接受数据并反馈，会导致主机误判断为没有执行成功，所以POLL\_RATE设置值在50ms以下，XY即为设置的间隔时间，单位ms。例如 AA 55 07 33 00 04 00 00 03 00指令即为设置查询间隔时间为30ms。

RESET指令如下：

|  |  |  |  |
| --- | --- | --- | --- |
| 起始符 | 长度 | 功能选择 | RESET的值 |
| AA 55（16进制） | 07 | 33 00 05 | 00 00 00 00 |

RESET指令使zigbee终端立即重启，与RESET的具体值无关，即RESET的值任意，推荐值00 00 00 00。例如AA 55 07 33 00 05 00 00 00 00指令立即使zigbee终端重启，与主机重新进行连接。

#### 3.1.7

可以通过串口指令对于电量计数进行清零，控制协议如下：

|  |  |  |  |
| --- | --- | --- | --- |
| 起始符 | 长度 | 功能选择 | 电量计数清零 |
| AA 55（16进制） | 07 | 33 00 06 | 00 00 00 00 |

电量计数即时清零，之后的电量从零开始重新计数，方便使用。

### 3．2 zigbee控制命令格式

#### 3.2.1

可以通过zigbee控制4路继电器通断状态，指令如下：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 起始符 | 长度 | 设备地址 | 控制命令 | 结束符 |
| a0 a1 a2 a3 a4（16进制） | 00 00 09 | aa 00 0x 55（x为具体地址） | 03 02 XX YY ZZ | b0 b1 b2 b3 b4 |

不同的zigbee终端设备以地址区分0x值为01~03，XX YY ZZ为具体控制继电器指令，XX为任意值，推荐00,YY为继电器状态值，YY值的2进制低4位分别代表4个继电器，1为开0为关，ZZ为校验和（ZZ = 03 + 02 + XX + YY + ff）,例如a0 a1 a2 a3 a4 00 00 09 aa 00 01 55 03 02 00 ff 03 b0 b1 b2 b3 b4为控制地址值为1的设备，4路继电器全开。

服务器端会收到返回值 a0 a1 a2 a3 a4 80 00 04 83 01 00 83 b0 b1 b2 b3 b4。

#### 3.2.2

可以通过读取电量模块的值来计数使用的电量，指令如下：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 起始符 | 长度 | 设备地址 | 控制命令 | 结束符 |
| a0 a1 a2 a3 a4（16进制） | 00 00 08 | aa 00 0x 55（x为具体地址） | 04 01 04 09 | b0 b1 b2 b3 b4 |

地址值0x为01~03，通过以上指令，服务器端即可接收到统计的消耗电量的值

服务器端收到指令协议如下：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 起始符 | 长度 | 功能说明 | 电量值 | 结束符 |
| a0 a1 a2 a3 a4（16进制） | 80 00 06 | 84 03 04 | XX YY ZZ | b0 b1 b2 b3 b4 |

电量值XX YY为16进制的电量，单位为0.1度，ZZ为校验和。消耗电量0.1度，则YY + 1，不到0.1度不会增加。（此处电量统计是K62模块2路测量电量消耗之和）例如接收到a0 a1 a2 a3 a4 80 00 06 84 03 04 7f 33 3c b0 b1 b2 b3 b4即为两路消耗累计值为0x7f33，即为消耗电量为3256.3度。

#### 3.2.3

可以通过指令读取当前4路继电器的通断状态，指令如下：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 起始符 | 长度 | 设备地址 | 控制命令 | 结束符 |
| a0 a1 a2 a3 a4（16进制） | 00 00 08 | aa 00 0x 55（x为具体地址） | 04 01 02 07 | b0 b1 b2 b3 b4 |

地址值0x为01~03，通过以上指令，可以读取继电器通断状态。

服务器端接收指令协议如下：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 起始符 | 长度 | 功能说明 | 继电器状态 | 结束符 |
| a0 a1 a2 a3 a4（16进制） | 80 00 06 | 84 03 02 | 00 0X YY | b0 b1 b2 b3 b4 |

继电器状态0X中X即为4路继电器的状态，X的4位2进制分别对应1到4号继电器的通断，1为开0为关，YY为校验和。例如接收到a0 a1 a2 a3 a4 80 00 06 84 03 02 00 09 91 b0 b1 b2 b3 b4即为1和4号继电器打开，2和3号继电器关闭。

#### 3.2.4

Zigbee的参数也可以使用zigbee来进行设置，但zigbee方式的设置不会立即使zigbee模块重启，设置的值需重启后生效，以避免误操作导致连接不上主机。

通过zigbee设置PANID（私网号）指令如下：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 起始符 | 长度 | 设备地址 | 控制命令 | 结束符 |
| a0 a1 a2 a3 a4（16进制） | 00 00 08 | aa 00 0x 55（x为具体地址） | 33 01 00 XX | b0 b1 b2 b3 b4 |

XX即为PANID的值（16进制），例如发送a0 a1 a2 a3 a4 00 00 08 aa 00 01 55 33 01 00 64 b0 b1 b2 b3 b4即为设置地址为1的zigbee终端PANID为100。

设置成功，返回值如下：

|  |  |  |  |
| --- | --- | --- | --- |
| 起始符 | 长度 | PANID设置成功 | 结束符 |
| a0 a1 a2 a3 a4（16进制） | 80 00 04 | 0x80 0x00 0x01 0xYY | b0 b1 b2 b3 b4 |

YY为校验和YY = 0x80 + 0x01 + 0xff（ff = aa + 55）

通过zigbee设置BAUD\_RATE（波特率）指令如下：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 起始符 | 长度 | 设备地址 | 控制命令 | 结束符 |
| a0 a1 a2 a3 a4（16进制） | 00 00 0a | aa 00 0x 55（x为具体地址） | 33 02 MM XX YY ZZ | b0 b1 b2 b3 b4 |

MM XX YY ZZ即为波特率的值（16进制），例如发送a0 a1 a2 a3 a4 00 00 0a aa 00 01 55 33 02 00 01 c2 00 b0 b1 b2 b3 b4即为设置zigbee地址为1的zigbee终端的波特率为115200。

设置成功，返回值如下：

|  |  |  |  |
| --- | --- | --- | --- |
| 起始符 | 长度 | BAUD\_RATE设置成功 | 结束符 |
| a0 a1 a2 a3 a4（16进制） | 80 00 04 | 0x80 0x00 0x02 0xYY | b0 b1 b2 b3 b4 |

YY为校验和YY = 0x80 + 0x02 + 0xff（ff = aa + 55）

通过zigbee设置UNI\_SEC\_ADDR（zigbee地址）指令如下：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 起始符 | 长度 | 设备地址 | 控制命令 | 结束符 |
| a0 a1 a2 a3 a4（16进制） | 00 00 08 | aa 00 0x 55（x为具体地址） | 33 03 XX YY | b0 b1 b2 b3 b4 |

XX YY即为zigbee地址，因为主机只能连接01~03号设备，所以XX为00，YY为01~03。例如发送a0 a1 a2 a3 a4 00 00 08 aa 00 01 55 33 03 00 02 b0 b1 b2 b3 b4为设置zigbee地址为1的zigbee终端的zigbee地址变更为02。

设置成功，返回值如下：

|  |  |  |  |
| --- | --- | --- | --- |
| 起始符 | 长度 | UNI\_SEC\_ADDR设置成功 | 结束符 |
| a0 a1 a2 a3 a4（16进制） | 80 00 04 | 0x80 0x00 0x03 0xYY | b0 b1 b2 b3 b4 |

YY为校验和YY = 0x80 + 0x03 + 0xff（ff = aa + 55）

通过zigbee设置POLL\_RATE（查询网络包间隔时间）指令如下：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 起始符 | 长度 | 设备地址 | 控制命令 | 结束符 |
| a0 a1 a2 a3 a4（16进制） | 00 00 08 | aa 00 0x 55（x为具体地址） | 33 04 0X 0Y | b0 b1 b2 b3 b4 |

因为要与主机及时通讯，所以POLL\_RATE要设置在50ms以内，XY代表时间，单位ms，例如发送a0 a1 a2 a3 a4 00 00 08 aa 00 01 55 33 04 03 00 b0 b1 b2 b3 b4即为设置zigbee地址为1的zigbee终端的POLL\_RATE为30，即30ms查询一次主机是否有网络数据包需要处理。

设置成功，返回值如下：

|  |  |  |  |
| --- | --- | --- | --- |
| 起始符 | 长度 | POLL\_RATE设置成功 | 结束符 |
| a0 a1 a2 a3 a4（16进制） | 80 00 04 | 0x80 0x00 0x04 0xYY | b0 b1 b2 b3 b4 |

YY为校验和YY = 0x80 + 0x04 + 0xff（ff = aa + 55）

通过zigbee控制终端RESET指令如下：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 起始符 | 长度 | 设备地址 | 控制命令 | 结束符 |
| a0 a1 a2 a3 a4（16进制） | 00 00 08 | aa 00 0x 55（x为具体地址） | 33 05 00 00 | b0 b1 b2 b3 b4 |

发送上述指令，控制相应zigbee地址的终端立即重启，以使之前的设置生效。例如发送a0 a1 a2 a3 a4 00 00 08 aa 00 01 55 33 05 00 00 b0 b1 b2 b3 b4即为使zigbee地址为1的终端立刻重启，重启后之前的设置生效。

设置成功，返回值如下：

|  |  |  |  |
| --- | --- | --- | --- |
| 起始符 | 长度 | 重启指令进行 | 结束符 |
| a0 a1 a2 a3 a4（16进制） | 80 00 04 | 0x80 0x00 0x05 0xYY | b0 b1 b2 b3 b4 |

YY为校验和YY = 0x80 + 0x05 + 0xff（ff = aa + 55）

#### 3.2.5

可以通过zigbee指令对电量计数进行清零，控制协议如下：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 起始符 | 长度 | 设备地址 | 控制命令 | 结束符 |
| a0 a1 a2 a3 a4（16进制） | 00 00 07 | aa 00 0x 55（x为具体地址） | 05 00 05 | b0 b1 b2 b3 b4 |

电量计数即时清零，并且会对主机返回电量计数为0的反馈，格式如下：

|  |  |  |  |
| --- | --- | --- | --- |
| 起始符 | 长度 | 电量为0的反馈 | 结束符 |
| a0 a1 a2 a3 a4（16进制） | 80 00 06 | 0x84,0x03,0x04,0x00,0x00,0x8a | b0 b1 b2 b3 b4 |

#### 3.2.6

通过zigbee对于 4路继电器初始状态进行设置，协议如下：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 起始符 | 长度 | 设备地址 | 控制命令 | 结束符 |
| a0 a1 a2 a3 a4（16进制） | 00 00 0b | aa 00 0x 55（x为具体地址） | 0a a2 00 XX XX XX XX | b0 b1 b2 b3 b4 |

其中0x为设备地址 后面的4个XX分别为4路继电器初始状态，00为关 01为开

设置成功返回值如下：

|  |  |  |  |
| --- | --- | --- | --- |
| 起始符 | 长度 | 继电器初始化设置成功 | 结束符 |
| a0 a1 a2 a3 a4（16进制） | 80 00 03 | 0x8a 0x00 0x89 | b0 b1 b2 b3 b4 |

## 4 其他测试

4个继电器状态指示灯亮灭情况随继电器通断而改变，无论通过手动拨继电器开关还是通过通讯协议改变继电器状态，LED均随之立刻变化。

### 

报警信号输入端口为设备面板上AL的2P端口，当有报警信号输入时，立刻进入报警模式，报警模式4路继电器的开关状态可以通过下述指令进行预先设定。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 起始符 | 长度 | 功能选择 | 设备地址 | 4路继电器状态 |
| AA 55（16进制） | 0A | B6 | 00 | d1 d2 d3 d4 XX XX XX XX |

表8 发送设置报警模式继电器状态命令格式

（长度为之后命令的字节数）

4路继电器状态的设置中，00为关，非0为开，XX为任意值。当设置成功时，串口会返回数据AA 55 03 6B 00 CC。

预先设置完成后，再次接收到报警信号，均会立即改变4个继电器状态到预设值。

## 5 可能问题分析以及注意事项

接下来对于测试过程中可能出现的一些问题，做简要分析。

初始化如果没有先对于POLL\_RATE进行设置，终端 无法即时反馈数据给主机，会造成返回值的错误和控制上的延迟。

对于zigbee的设置应先使用串口进行测试，以确定了解zigbee的PANID，BAUDRATE，UNI\_SEC\_ADDR等参数，以免使用zigbee时进行错误操作。

Zigbee主机与终端之间有时会进行自动连接，即右侧LED灯闪烁，此时的zigbee控制指令会暂时无法正常控制，待重新连接完成后，即可正常工作。确保测试在zigbee连接完成的情况下进行。

串口控制如果控制动作正常 ，没有返回值，除了RS232电路部分可能有误以外，也有可能是USB转RS232转换器与电脑之间连接不稳定造成的。