

**D180与上位机**

**通讯链路层**

**协议格式**

[V1.00]

|  |
| --- |
| PAX Computer Technology(Shenzhen) Co., LTD  Tel: +86-755-86169630  Fax: +86-755-86169634  Address:4/F,No.3 Building, Software Park, Second Central Science-Tech Road, High-Tech Industrial Park, Shenzhen, Guangdong, P.R. China  Website. <http://www.pax.com.cn> |

# Document Changes

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Ver. | Description | Author |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# Contents

[Document Changes 2](#_Toc385945473)

[Contents 3](#_Toc385945474)

[Tables 4](#_Toc385945475)

[1 链路层通讯协议 5](#_Toc385945476)

[1.1 通讯模式 5](#_Toc385945477)

[1.1.1 单帧 5](#_Toc385945478)

[1.1.2 多帧 5](#_Toc385945479)

[1.2 通讯包格式 6](#_Toc385945480)

[1.2.1 包序号 6](#_Toc385945481)

[1.2.2 帧序号 6](#_Toc385945482)

[1.2.3 确认帧 6](#_Toc385945483)

[1.2.4 同步/握手帧 6](#_Toc385945484)

[1.2.5 失步帧 7](#_Toc385945485)

[1.3 重发 7](#_Toc385945486)

[1.4 失步处理约定 8](#_Toc385945487)

[1.5 包大小和分包规则 8](#_Toc385945488)

# Tables

[Table 1单帧通讯 5](#_Toc385944997)

[Table 2多帧通讯 5](#_Toc385944998)

[Table 3 通讯包格式定义 6](#_Toc385944999)

[Table 4 ACK 帧 6](#_Toc385945000)

[Table 5 NAK帧 6](#_Toc385945001)

[Table 6 同步帧 7](#_Toc385945002)

[Table 7 同步请求帧 7](#_Toc385945003)

[Table 8 同步响应帧 7](#_Toc385945004)

[Table 9失步帧 7](#_Toc385945005)

# 链路层通讯协议

## 通讯模式

适用于上位机和下位机采用一发一收的通讯模式。上位机发送请求报文，下位机接收并处理命令后，发送响应报文。

请求报文和响应报文在传送时，发送方发送一帧命令数据帧后，接收方就回送一个确认帧，确认帧包括接收（ACK）帧或拒绝(NAK)帧。ACK帧表示接收方确认收到报文。NAK帧表示报错，发送收到拒绝帧NAK后，需要重发。

当被传输数据的长度超过一帧最大限制时，可以多帧传输。帧的协议尾设为0x17，表明还有后续帧。

接收方处理完命令后，回复一个响应报文，返回执行的结果。

### 单帧

|  |  |  |
| --- | --- | --- |
| 发送方（上位机） |  | 接收方（下位机） |
| 请求报文 | 🡪🡪🡪 |  |
|  | 🡨🡨🡨 | **ACK** |
|  | 🡨🡨🡨 | **应答报文** |
| ACK | 🡪🡪🡪 |  |

Table 1单帧通讯

### 多帧

|  |  |  |
| --- | --- | --- |
| 发送方（上位机） |  | 接收方（下位机） |
| 请求报文1st 帧 | 🡪🡪🡪 |  |
|  | 🡨🡨🡨 | **ACK** |
| 请求报文2nd 帧 | 🡪🡪🡪 |  |
|  | 🡨🡨🡨 | **ACK** |
| 请求报文nth 帧 | 🡪🡪🡪 |  |
|  | 🡨🡨🡨 | **ACK** |
|  | 🡨🡨🡨 | **响应报文1st 帧** |
| ACK | 🡪🡪🡪 |  |
|  | 🡨🡨🡨 | **响应报文2nd 帧** |
| ACK | 🡪🡪🡪 |  |
|  | 🡨🡨🡨 | **响应报文nth 帧** |
| ACK | 🡪🡪🡪 |  |

Table 2多帧通讯

## 通讯包格式

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **STX** | **版本** | **包序号（PaNo）** | | **帧序号（FrNo）** | | **数据域长度Len** | | **数据域** | **LRC** | **ETX** |
| PaNoH | PaNoL | FrNoH | FrNoL | LenH | LenL |  |
| 起始符固定为0x02 | 包版本号，当前固定为0x01 | PaNoH=PaNo/258  PaNoL=PaNo%256  从1000开始。 | | 从1开始，  PaNoH=PaNo/258  PaNoL=PaNo%256 | |  | |  | 从STX开始全部数据LRC校验值 | 结束符：  0x03结束帧  0x17有后续帧 |

Table 3 通讯包格式定义

### 包序号

包序号由上位机维护，用于标识请求包，从1000开始，1000以下用于表示控制帧（确认帧ACK/NAK，同步帧等）。

上位机每次发送报文时加1。下位机响应报文保持包序号。

如果接收报文和响应报文的包不同步，则上位机需要先发送同步帧后再发请求报文。

### 帧序号

帧序号用于标识多帧通讯的帧序列，帧序号从1开始，由发送方每发一帧后累加。

### 确认帧

#### ACK帧

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **STX** | **版本** | **包序号（PaNo）** | | **帧序号（FrNo）** | | **数据域长度Len** | | **数据域** | **LRC** | **ETX** |
| PaNoH | PaNoL | FrNoH | FrNoL | LenH | LenL |  |
| 0x02 | 0x01 | 0x00,0x00 | |  | | 0x00,0x01 | | 0X06 | LRC | 0x03 |

Table 4 ACK 帧

#### NAK帧

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **STX** | **版本** | **包序号（PaNo）** | | **帧序号（FrNo）** | | **数据域长度Len** | | **数据域** | **LRC** | **ETX** |
| PaNoH | PaNoL | FrNoH | FrNoL | LenH | LenL |  |
| 0x02 | 0x01 | 0x00,0x00 | |  | | 0x00,0x01 | | 0X15 | LRC | 0x03 |

Table 5 NAK帧

### 同步/握手帧

|  |  |  |
| --- | --- | --- |
| 发送方 |  | 接收方 |
| 同步请求帧 | 🡪🡪🡪 |  |
|  | 🡨🡨🡨 | **ACK** |
|  | 🡪🡪🡪 | 同步响应帧 |
| ACK | 🡪🡪🡪 |  |

Table 6 同步帧

#### 请求帧

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **STX** | **版本** | **包序号（PaNo）** | | **帧序号（FrNo）** | | **数据域长度Len** | | **数据域** | **LRC** | **ETX** |
| PaNoH | PaNoL | FrNoH | FrNoL | LenH | LenL |  |
| 0x02 | 0x01 | 0x00,0x01 | | 0x00，0x01 | | 0x00,0x00 | | 无 | LRC | 0x03 |

Table 同步请求帧

#### 响应帧

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **STX** | **版本** | **包序号（PaNo）** | | **帧序号（FrNo）** | | **数据域长度Len** | | **数据域** | **LRC** | **ETX** |
| PaNoH | PaNoL | FrNoH | FrNoL | LenH | LenL |  |
| 0x02 | 0x01 | 0x00,0x01 | | 0x00，0x01 | | 0x00,0x08 | | PaSize+ FrSize | LRC | 0x03 |

Table 同步响应帧

包大小PaSize：下位机支持的包最大长度，4字节，大端。

帧大小FrSize：下位机支持的帧最大长度值，4字节，大端

### 失步帧

下位机侦测失步时，丢弃请求报文，直接响应失步帧。

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **STX** | **版本** | **包序号（PaNo）** | | **帧序号（FrNo）** | | **数据域长度Len** | | **数据域** | **LRC** | **ETX** |
| PaNoH | PaNoL | FrNoH | FrNoL | LenH | LenL |  |
| 0x02 | 0x01 | 0x00,0x02 | | 0x00，0x01 | | 0x00,0x00 | | 无 | LRC | 0x03 |

Table 失步帧

## 重发

重发包括超时重发和NAK重发，

发送方在150ms内如果没有收到确认帧，将重发送当前帧。

发送方如果收到NAK帧，将重新发送当前帧。

如果接收房收到到重发帧（帧序号比当前预期值小1的帧数据包），则将当前数据包丢弃。

重发依然失败后，则返回错误给调用层。

## 失步处理约定

失步包括以下两种场景：

对于上位机，若响应包与请求包的包序号不相等；

对于下位机，若请求包序号不等于当前期望序列号或者帧序号不是当前预期值并且不是不是比当前期望帧序号小1。

失步的处理以上位机为主体来控制。

上位机检测到失步时，向下位机下发同步请求包。下位机重置包序号（同步包序号），并响应帧大小和包大小。

座机上检测到失步时，直接响应失步帧。

## 包大小和分包规则

包大小和分包规则（即帧大小）通过同步帧，下位机告知上位机。分包大小以下位机的帧最大值为准。目前D180蓝牙通讯帧大小为1000字节。USB自带有流控，因此帧大小定为1000字节。

包大小涉及到应用层通讯包及全局变量的分配。需要应用预先协商定义。