|  |
| --- |
|  |

|  |
| --- |
| **蓝牙串口通信协议** |
|  |
|  |
| 基于串通信的一些基本协议规范及说明 |
|  |
|  |
| **2015/9/10** |
|  |

目录

[1、 协议说明 3](#_Toc430529958)

[1.1、 协议物理层特性 3](#_Toc430529959)

[1.2、 协议注意事项 3](#_Toc430529960)

[2、 协议格式 4](#_Toc430529961)

[2.1、基本格式说明 4](#_Toc430529962)

[2.2、命令内容格式 4](#_Toc430529963)

[2.3、CRC校验说明 4](#_Toc430529964)

[2.3.1、确定多项式 5](#_Toc430529965)

[2.3.2程序算法 5](#_Toc430529966)

[3、 命令详细说明 7](#_Toc430529967)

[4、 外设（功能）编码详细说明 8](#_Toc430529968)

[4.1、腕表锁定状态编码 8](#_Toc430529969)

[4.2、蓝牙辅助定位状态编码 8](#_Toc430529970)

[4.3、远程拍照状态编码 8](#_Toc430529971)

[4.4、腕表反向查找状态编码 8](#_Toc430529972)

[4.5、腕表绑定激活状态编码 8](#_Toc430529973)

[4.6腕表连接验证编码 9](#_Toc430529974)

[4.7、距离传感器编码 9](#_Toc430529975)

[4.8、重力传感器编码 9](#_Toc430529976)

|  |  |  |
| --- | --- | --- |
| 作者 | 说明 | 日期 |
| 熊亮 | 初稿 | 2015-09-10 |
| 熊亮 | 修改协格式 | 2015-09-19 |
|  |  |  |

# 协议说明

## 协议物理层特性

本协议物理层特性如下：

通信方式： UART

Baud rate： 115200bps

Start bit: 1bit：0

Stop bit： 1bit：1

Data bit： 8bit

Parity bit： No

发送时序： LSB->MSB

## 协议注意事项

在通信中，应注意以下几点：

1. 通讯中ARM系统（手表系统）为Master，蓝牙系统为 Slave。
2. 命令方向均以ARM系统定义，W为写，即ARM控制蓝牙设备；R为读，即ARM获得蓝牙相关状态；PR为被动读，即蓝牙主动上报相关状态。
3. 有些重要数据包传输后，需要等待对方回复对应消息确认，如果500MS（特殊情况，会定义回复超时时间）没等到该消息，则继续发送该数据包，以确保通讯的正确性。

# 协议格式

## 2.1、基本格式说明

本文档定义的所有协议遵循如下的格式，如表1－1所示。

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 名称 | 同步字 | 命令数量 | 长度 | 命令内容 | CRC |
| 字节数/Byte | 2 | 1 | 1 | N | 2 |

表1－1协议基本格式

**同步字：**用于数据包同步，本协议指定同步字为0x5AA5，接收方收到相同的同步字后才认为是有效的数据包。

**命令数量：** 此数据包所包含的命令的数目。

**长度字：** 命令内容字节数（N）单位为Bytes。

**命令内容：** 命令信息。格式见3.2 命令内容格式介绍

**CRC：** 命令校验。同步字、命令数量、长度、命令内容，即整包数据的差错校验，关于本协议所用的CRC校验算法见2.3节CRC校验说明。

**注：由于考虑到蓝牙芯片的RAM限制，这里规定2+1+1+N+2<=512Bytes**

## 2.2、命令内容格式

命令内容的格式如表1-2所示。

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 名称 | Cmd1  Len | Cmd1  Dev | Cmd1  Code | Cmd1  Info | … | CmdN  Len | CmdN  Dev | CmdN  Code | CmdN  Info |
| 字节数 | 1 | 1 | 1 | N | … | 1 | 1 | 1 | N |

表1-2 命令内容格式

**Cmd1Len：** 命令1的长度 = Cmd1Dev字节数（1） + Cmd1Code字节数（1）+ Cmd1Info字节数（N），单位Bytes

**Cmd1Dev：** 命令1的控制外设（功能）编号

**Cmd1Code：** 命令1外设（功能）编码

**Cmd1Info：** 命令1信息字

**…**

**CmdNLen：** 命令N的长度 = CmdNDev字节数（1） + CmdNCode字节数（1）+ CmdNInfo字节数（N），单位Bytes

**CmdNDev：** 命令N的控制外设（功能）号

**CmdNCode：** 命令N外设（功能）编码

**CmdNInfo：** 命令N信息字

## 2.3、CRC校验说明

CRC（循环冗余校验码）：CRC 算法的基本思想是将传输的数据当做一个位数很长的数。将这个数除以另一个数。得到的余数作为校验数据附加到原数据后面。传输的时候将传输的数据和校验数据一块发送这个，在接收端我们利用接收到的数据利用“模二除法”除以利用的多项式，如果余数为0说明传输过程中没有差错，如果不为0表明传输过程中有错误。

### 2.3.1、确定多项式

通常我们会采用固定的多项式，常见的几种生成多项式如：

CRC8=X8+X5+X4+X0

CRC-CCITT=X16+X12+X5+X0

CRC16=X16+X15+X2+X0

CRC12=X12+X11+X3+X2+X0

CRC32=X32+X26+X23+X22+X16+X12+X11+X10+X8+X7+X5+X4+X2+X1+X0

生成多项式确定之后我们就可以进行算法编程。从前面的介绍我们知道CRC校验核心就是实现无借位的除法运算（即模二除法）。

常见的三种CRC 标准用到个各个参数如表1－3所示。

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | CCITT | CRC16 | CRC32 |
| 校验和位宽W | 16 | 16 | 32 |
| 生成多项式 | X16+X12+X5+X0 | X16+X15+X2+X0 | X32+X26+X23+X22+X16+X12  +X11+X10+X8+X7+X5+X4+X2+X1+X0 |
| 除数（多项式） | 0x1021 | 0x8005 | 0x04C11DB7 |
| 余数初始值 | 0xFFFF | 0x0000 | 0xFFFFFFFF |
| 结果异或值 | 0x0000 | 0x0000 | 0xFFFFFFFF |

表1－3常见CRC参数

### 2.3.2程序算法

加入这些变形后，常见的算法描述形式就成了这个样子了：

（1） 设置CRC寄存器，并给其赋值为“余数初始值”。

（2） 将数据的第一个8-bit字符与CRC寄存器进行异或，并把结果存入CRC寄存器。

（3） CRC寄存器向右移一位，MSB补零，移出并检查LSB。

（4） 如果LSB为0，重复第三步；若LSB为1，CRC寄存器与0x31相异或。

（5） 重复第3与第4步直到8次移位全部完成。此时一个8-bit数据处理完毕。

（6） 重复第2至第5步直到所有数据全部处理完成。

（7） 最终CRC寄存器的内容与“结果异或值”进行或非操作后即为CRC值。

*U16 crc16\_check(const U8 \*data\_ptr, U8 data\_length)*

*{*

*/\*polynomial\*/*

*U16 crc\_gen = 0x1021;*

*U16 crc;*

*U8 i, j;*

*/\*init value of crc\*/*

*crc = 0xFFFF;*

*if (data\_length != 0)*

*{*

*for (i = 0; i < data\_length; i++)*

*{*

*crc ^= (U16)(data\_ptr[i]);*

*for (j = 0; j < 8; j++)*

*{*

*if ((crc & 0x01) == 0x01)*

*{*

*crc >>= 1;*

*crc ^= crc\_gen;*

*}*

*else*

*{*

*crc >>= 1;*

*}*

*}/\*end for\*/*

*}/\*end for\*/*

*}/\*end if\*/*

*return crc;*

*}*

# 命令详细说明

控制外设（功能）编号如表1－4所示。

|  |  |  |
| --- | --- | --- |
| 控制外设（功能）编号 | 描述 | 方向 |
| 0x01 | 锁定腕表 | W |
| 0x02 | 蓝牙辅助定位 | W/PR |
| 0x03 | 远程拍照 | W |
| 0x04 | 反向查找 | W |
| 0x05 | 绑定激活 | W/R |
| 0x06 | 连接验证 | W/R |
| 0x07 | 距离传感器数据上报 | PR |
| 0x08 | 重力传感器数据上报 | W/PR |
| …… | …… | …… |

表1－4控制外设（功能）编号

# 外设（功能）编码详细说明

## 4.1、腕表锁定状态编码

腕表锁定状态编码如表1－5所示。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 命令编码 | 命令方向 | 描述 | 命令信息字长度 | 备注 |
| 0x00 | W | 腕表解除锁定状态 | 无 | 无需回复 |
| 0x01 | W | 腕表进入锁定状态 | 15bytes IMEI号 | IMEI为ASCII码 |

表1－5腕表锁定状态编码

## 4.2、蓝牙辅助定位状态编码

腕表蓝牙辅助定位编码如表1－6所示。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 命令编码 | 命令方向 | 描述 | 命令信息字长度 | 备注 |
| 0x00 | W | 腕表解除蓝牙辅助定位 | 无 | 无需回复 |
| 0x01 | W | 腕表允许蓝牙辅助定位 | 无 | 需回复 |
| PR | 蓝牙辅助定位信息 | 1byte | RSSI值 |

表1－6腕表蓝牙辅助状态定位

## 4.3、远程拍照状态编码

腕表远程拍照编码如表1－7所示。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 命令编码 | 命令方向 | 描述 | 命令信息字长度 | 备注 |
| 0x00 | W | 腕表退出远程拍照 | 无 | 无需回复 |
| 0x01 | W | 腕表按下远程拍照按键 | 无 | 无需回复 |

表1－7腕表远程拍照状态编码

## 4.4、腕表反向查找状态编码

腕表反向查找编码如表1－8所示。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 命令编码 | 命令方向 | 描述 | 命令信息字长度 | 备注 |
| 0x00 | W | 腕表退出反向查找 | 无 | 无需回复 |
| 0x01 | W | 腕表进入反向查找状态 | 无 | 无需回复 |

表1－8腕表反向查找状态编码

## 4.5、腕表绑定激活状态编码

腕表绑定激活状态编码如表1－9所示。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 命令编码 | 命令方向 | 描述 | 命令信息字长度 | 备注 |
| 0x00 | W | 腕表退出绑定激活状态 | 无 | 无需回复 |
| 0x01 | W | 腕表进入绑定激活状态 | 无 | 无需回复 |

表1－9腕表绑定激活状态编码

## 4.6腕表连接验证编码

腕表连接验证编码如表1－10所示。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 命令编码 | 命令方向 | 描述 | 命令信息字长度 | 备注 |
| 0x00 | W | 腕表发送验证随机数 | Byte0,byte1:0x5E5E  Byte2-byte5:4字节随机数  6bytes | 需回复 |
| R | 腕表接收手机端传回的随机数加密运算数据 | 经过加密运算的4字节加密运算数据 | Randomdata ^ 0x1A2B3C4D |
| 0x01 | W | 验证通过，腕表发送IMEI | Byte0,byte1:0x3C3C  Byte2-byte16:IMEI号  17bytes | 无需回复 |

表1－10腕表连接验证编码

## 4.7、距离传感器编码

## 4.8、重力传感器编码