**说明**：由于新的app上暂时没有用户的概念，以后有了服务器之后才能加上，所以暂时不加用户。

1. 关于数据的存储问题：板子端存储密码、指纹、射频卡的数据，以及他们对应的编号，（如果内存允许的话，也可以考虑直接存汉字）。
2. 关于指纹、密码、射频卡需要存储的内容：

板子端存储：

① 密码的存储：首先是该密码的编号（编号定在3位，如：001），密码的内容（暂时定在6位）。密码的有效时间，允许存储的密码个数根据板子端来定。

②指纹的存储：该指纹的编号，指纹的数据。

③射频卡的存储：射频卡的编号（3位），射频卡的数据。

App端存储：

①密码的存储：手机端存储该密码的名称（如xx的密码），密码的编号（主要是为了删除该密码用）

②指纹的存储：手机端存储该指纹的编号、名称。

③射频卡存储：手机端存储射频卡的编号、名称。

**具体流程：（数据包的格式按照之前的不变）**

1. **录指纹流程：**

1.首先，手机端发送录指纹命令通知板子即将录指纹，命令为（0x0301）数据包格式如下：

|  |  |
| --- | --- |
| magicCode | 0xfecf |
| version | 0x0001 |
| totalLength | 12 |
| cmdId | 0x0301 |
| seq |  |
| errorCode | 0x0000 |
| 包体 | 无 |

1. 板子端回包：errorCode: 1、OK（0x0000）,2、异常（0x0800）

（1）异常时退出录指纹界面，板子端退出录指纹函数。

（2）板子端回包OK之后接下来手机端显示录三次指纹操作，板子端给与相应的语音提醒。

第一次录指纹：

板子回包：errorCode：1、OK（0x0000），2、采集超时：（0x0500），3、指纹质量差（0x0300）

①当出现2、3的情况之后，板子端退出录指纹流程，app退出录指纹界面，

②第一次录取成功之后，数据包中需要将seq序列号置00 01，表示第一次录入成功，app上根据该序列号显示是第几次录入成功。手机端不回复板子，接着第二次录指纹。

第二次录指纹：

板子回包：errorCode：1、OK（0x0000），2、采集超时：（0x0500），3、指纹质量差（0x0300）

①当出现2、3的情况之后，板子端退出录指纹流程，app退出录指纹界面，如果想再录指纹的话，需要重新启动录指纹流程。

②第二次录取成功之后，数据包中需要将seq序列号置00 02接着第三次录指纹。

第三次录指纹：

板子回包：errorCode：1、OK（0x0000）, 2、采集超时：（0x0500），3、指纹质量差（0x0300）

⑴当出现2、3的情况之后，板子端退出录指纹流程，app退出录指纹界面，如果想再录指纹的话，需要重新启动录指纹流程。

⑵当录入成功之后，板子端需要将生成的指纹编号发送给手机端，数据包格式如下：

此时将seq的值写成00 03

|  |  |
| --- | --- |
| magicCode | 0xfecf |
| version | 0x0001 |
| totalLength | 14 |
| cmdId | 0x0301 |
| seq | 00 03 |
| errorCode | 0x0000 |
| 包体 | 见下文 |

包体(3B)

|  |  |
| --- | --- |
| Index | 内容 |
| 0~2 | 指纹序号0x303131（011） |

**说明：seq的值在录指纹的时候写入值，其他录密码以及录射频卡流程不需要写值，默认写成00 00**

**2.录射频卡流程：**

1.首先app进入录射频卡界面，此时点击录卡按钮，手机向板子发送录卡指令（0x0302）

数据包如下：

|  |  |
| --- | --- |
| magicCode | 0xfecf |
| version | 0x0001 |
| totalLength | 12 |
| cmdId | 0x0302 |
| seq |  |
| errorCode | 0x0000 |
| 包体 | NULL |

2.接着板子端向手机回包：

可能的应答：1.OK(0x0000);2采集超时（0x0500）

1. 采集超时，板子退出录入流程，app退出录入界面，再录入时需要重新开始该流程。
2. OK时，说明此时已经采集成功，板子需要将该射频卡的编号发送给app。

数据包格式如下：

|  |  |
| --- | --- |
| magicCode | 0xfecf |
| version | 0x0001 |
| totalLength | （12+包体） |
| cmdId | 0x0302 |
| seq | 00 00 |
| errorCode | 0x0000 |
| 包体 | 如下： |

包体(3B)

|  |  |
| --- | --- |
| Index | 内容 |
| 0~2 | 射频卡编号0x303131（011） |

**3.录密码流程：**

1.首先app进入到录密码界面，点击录入按钮之后开始录入密码，录密码命令（0x0104）

数据包格式：

|  |  |
| --- | --- |
| magicCode | 0xfecf |
| version | 0x0001 |
| totalLength | （12+包体） |
| cmdId | 0x0104 |
| seq | 00 00 |
| errorCode | 0x0000 |
| 包体 | 见下文 |

包体内容：

|  |  |
| --- | --- |
| Index | 内容 |
| 0~2 | 编号(例如020)（0x30 32 30） |
| 3~9 | 密码例如：123456(0x31 32 33 34 35 36) |
| 9~ | 密码准入时间 |

说明：准入时间可以选择为24小时、7天、永久、自定义时间（手机端选择）：

可以由totalLength的长度判断是不是永久密码！

（1）24小时、7天、自定义时间。如下格式：（此时包体加上包头一共为12+9+24B）

2018年1月16日17时50分表示为:

年 月/日 时/分

起始：0x32 0x30 0x31 0x38 0x30 0x31 0x31 0x36 0x31 0x37 0x35 0x30 12字节

2018/01/16 17:50

终止：0x32 0x30 0x31 0x38 0x30 0x31 0x31 0x37 0x31 0x37 0x35 0x30 12字节

2018/01/17 17:50

（2）如果是永久的话，准入时间字段不写（此时手机端直接发送密码编号）

此时的包体加上包头的长度为12+9个字节



2.板子收到数据之后，回包给app，此时不需要携带包体

出现的应答如下：errorCode: 1、OK（0x0000）,2、异常（0x0800）

ok，说明保存成功

异常时板子和app都退出录密码流程。下次再录需要重新进入该流程

**4．删除指纹流程：**

1. 首先app端进入删除指纹界面，点击删除某一个指纹，删除指令为（0x0802），

数据包如下：

|  |  |
| --- | --- |
| magicCode | 0xfecf |
| version | 0x0001 |
| totalLength | 12+包体3 |
| cmdId | 0x0802 |
| seq |  |
| errorCode | 0x0000 |
| 包体 | 见下文 |

包体

|  |  |
| --- | --- |
| Index | 内容 |
| 0~2 | 用户编号(例如020)（0x30 31 30） |

2.板子端回包：命令不变，包体没有

可能的应答：1、OK（0x0000）,2、异常（0x0800）

1. ok时说明已经删除。
2. 异常时退出删除流程。

**5．删除射频卡流程：**

1. App进入删除射频卡界面，点击删除某一个射频卡，删除指令为（0x0803）

数据包格式如下：

|  |  |
| --- | --- |
| magicCode | 0xfecf |
| version | 0x0001 |
| totalLength | 15 |
| cmdId | 0x0803 |
| seq | 00 00 |
| errorCode | 0x0000 |
| 包体 | 见下文 |

包体内容（3B）

|  |  |
| --- | --- |
| Index | 内容 |
| 0~2 | 编号(例如020)（0x30 32 30） |

2.板子端回包：命令不变，包体没有

可能的应答：1、OK（0x0000）,2、异常（0x0800）

1. ok时说明已经删除。
2. 异常时退出删除流程。

**5．删除密码流程：**

1.App向板子发送删除密码指令：（0x0804）

数据包格式：

|  |  |
| --- | --- |
| magicCode | 0xfecf |
| version | 0x0001 |
| totalLength | 15 |
| cmdId | 0x0804 |
| seq |  |
| errorCode | 0x0000 |
| 包体 | 见下文 |

包体内容（3B）

|  |  |
| --- | --- |
| Index | 内容 |
| 0~2 | 编号(例如020)（0x30 32 30） |

2.板子端回包：命令不变，包体没有

可能的应答：1、OK（0x0000）,2、异常（0x0800）

1. ok时说明已经删除。
2. 异常时退出删除流程。

**6.一键开锁**

**首先手机连接上蓝牙之后，进入主界面如图所示：**

**一键开锁指令（0x0106）**

**此时app发送如下数据包给板子**

|  |  |
| --- | --- |
| magicCode | 0xfecf |
| version | 0x0001 |
| totalLength | 12 |
| cmdId | 0x0106 |
| seq | 00 00 |
| errorCode | 0x0000 |
| 包体 | 无 |

**板子回包：1.ok(0x0000) 2.(0x0800)异常**

**回复Ok时，app显示已开锁，异常时显示未开锁。**

**门锁关闭：（当没有手机和板子相连的话，不用发以下指令）**

**当门锁关闭之后板子端需要发送相应数据包给手机，手机显示门锁已关闭，板子的发包为：指令为(0x0108)**

|  |  |
| --- | --- |
| magicCode | 0xfecf |
| version | 0x0001 |
| totalLength | 12 |
| cmdId | 0x0108 |
| seq | 00 00 |
| errorCode | 0x0000 |
| 包体 | 无 |

手机端收到数据后不回包给板子，板子控制门锁关闭就行

**7.关于手机录入操作过程中的一些突发情况。**

**一．蓝牙连接良好，但是用户想在该界面退出时情况：**

1. **录指纹情况：（录指纹界面有两个，其中一个是三次录入时的显示界面，三次录入完成之后，板子端立即保存数据，但是app端会跳转到需要用户输入指纹名字的界面，所以分成以下情况）**

**①当app通知板子录指纹时，板子回复ok进入三次录指纹界面，回复异常时，两边同时退出。**

**②当app进入三次录指纹界面时，在第一次以及第二次录入成功过程中，如果用户突然想退出录指纹，这时app发送异常（0x0800）数据包如下：**

|  |  |
| --- | --- |
| magicCode | 0xfecf |
| version | 0x0001 |
| totalLength | 12 |
| cmdId | 0x0301 |
| seq | 00 00 |
| errorCode | 0x0800 |
| 包体 | 无 |

**App退出录指纹流程，板子端退出录指纹函数。**

**③当指纹第三次已经采集成功，板子端已经将编号数据发过来，如果此时app端想按手机返回键退出该流程，因为此时板子端已经将编号、指纹数据保存，所以app不能退出只能保存之后退出。就是说当第三次已经录入成功之后，app端不能直接退出，必须将对该指纹名称命名然后保存（该操作app端可以实现，板子端不用管）**

**（2）录密码情况：因为录密码界面只有一个，和录指纹界面不同。**

**当用户在该界面按手机返回键退出该录入界面时，app发送退出指令（0x0800）异常app退出录入，数据包如下：**

|  |  |
| --- | --- |
| magicCode | 0xfecf |
| version | 0x0001 |
| totalLength | 12 |
| cmdId | 0x0104 |
| seq | 00 00 |
| errorCode | 0x0800 |
| 包体 | 无 |

**板子端收到数据包之后根据errorcode（0x0800）退出录入指纹函数。**

**（3）录射频卡情况。（录射频卡和录指纹相似，但是录射频卡没有录三次流程）**

**当录入成功之后，app转跳界面，跳到输入该射频卡的名字界面。此时板子端已经将射频卡数据保存，编号已经发出。如果此时用户想按手机返回键退出，此时提示用户必须输入名字保存之后才能退出（app端做板子端不用管）**

**二．在一些录入操作的过程中蓝牙突然中断情况：**

**因为app端进行录入操作都是基于蓝牙的连接，如果蓝牙中断，板子和app两端都必须要考虑这一情况。如果细分起来的话情况比较多，比如：录入之前蓝牙断开，还是录入过程中断开，还是录入成功之后蓝牙断开都需要考虑。蓝牙断开之后，因为板子和手机不能通信，两端必须加入自行判断退出机制。**

**总的来说，不论蓝牙处于什么时候中断，板子和手机都必须自行退出到最初的函数，或者界面。板子端可以通过判断蓝牙的连接状态判断是不是有手机和自己相连，如果没有相连，退出当前的函数，手机端也会根据蓝牙的状态来做出相应的退出。**

**板子端在每次接收到数据之后判断此时是不是有手机相连，如果没有手机相连的话，不保存数据，而且要退出相应的流程，app端也会退出相应界面**