# 无线换装上位机软件总结

1. 目的

本文档记录了在开发无线换装上位机软件过程中的问题与总结，用以后续扩展开发总结经验教训。

1. 问题及解决
2. 协议数据定义格式

解决：最初考虑使用json，但json与python数据转换类型少，无法满足实际情况；后改用python3中的Struct模块打包和解包C语言中的结构体,及二进制数据的处理;

1. 数据存储不当，导致上位机与主机不断重连；

问题原因：调试过程中发现，串口数据会存在切片，导致收到的数据包不完整；

解决：排除主机发送数据不完整后，定位是上位机软件读物数据时，一次无法读取到全部包数据；使用多线程处理，读取线程不断读取数据；主线程处理数据时，同步更新缓冲区数据大小，避免有效数据切片。同时新建定时发送数据线程，防止应无法接收有效周期包而产生重连，导致其他业务流程无法继续（调试使用，正常开发不可采取）;代码重构后：使用数据池，解决丢包问题。

1. 调试信息不完整，导致定位主机问题困难。

解决：重定向关键调试信息至文件，快速定位问题

1. 全局变量管理使用不当，导致维护困难

解决：定义一个全局变量字典，用来维护全部变量

1. *随着测试进行，因代码设计原因，导致无法满足VV异常测试。(需要创建大量的线程，才能满足异常测试需要)*

*解决：重构代码；*

*划分模块：通信层(数据的收发)、数据预处理层（数据的转译与解译）、数据管理(数据的解析与组包)、业务处理层（业务流程处理）*

**

1. *为方便初期调试，部分数据采用写死的方法写入代码，灵活性低。（如文件CRC等）*

*解决：从文件中获取相关信息，或计算文件相关信息。*

*7.模块引用问题：*

*例如：串口模块不能重复安装，安装前需完全卸载*

#报错：module ‘serial’ has no attribute ‘Serial’

#解决办法：1、卸载serial；2、卸载pyserial；3、重新打开你的编辑器。

#pip uninstall serial

#pip uninstall pyserial

#pip install pyserial

三：总结经验教训：

1. 先梳理清楚需求，协议等。
2. 先搞好设计，不要急于实现；设计时，需请教有经验的人，避免做无用功。
3. 编码过程中，及时更新提交代码，注释明白，方便维护修改。
4. 开发过程中，需要考虑异常测试接口预留及调试信息输出。